



PDF hosted at the Radboud Repository of the Radboud University Nijmegen

The following full text is a publisher's version.

For additional information about this publication click this link.

<http://hdl.handle.net/2066/17225>

Please be advised that this information was generated on 2017-12-05 and may be subject to change.

D.7

*The Type Free Lambda Calculus**

HENK P. BARENDREGT**

Contents

0. Introduction	1092
1. Towards the theory	1094
2. Classical λ -calculus	1100
3. Construction of $\mathcal{P}\omega$	1106
4. Construction of D_∞	1110
5. Solvability	1117
6. Böhm trees	1119
7. Analysis of D_∞	1125
References	1131

* Part of the chapter was written while visiting the Forschungsinstitut für Mathematik, ETH, Zurich.

** The author wishes to thank Dana Scott for his basic remarks on a draft of this chapter, and Jane Bridge and Jeff Zucker for pointing out several errors in the text.

0. Introduction

The λ -calculus and its variable free equivalent, combinatory logic, were initiated around 1930 by Church, Schönfinkel and Curry respectively. The intention of the founders of the subject was to study *rules*; in other words to study the old-fashioned notion of “function” in the sense of *definition*. In contrast to Dirichlet’s notion (of *graph*, that is the set of pairs of argument and associated value) the older notion referred to the *process* of stepping from argument to value, a process coded by a definition. Generally we think of such definitions as given by words in ordinary English, applied to arguments also expressed by words (in English). Or, more specifically, we may think of the definitions as programs for machines applied to, that is, operating on, such programs. In both cases we have to do with a *type free* structure, where the objects of study are at the same time function and argument. In particular, a function can be applied to itself. For the usual conception of a function in mathematics (in Zermelo–Fraenkel set theory) this is impossible (because of the axiom of foundation).

The λ -calculus represents a class of (partial) functions (λ -definable functions) on the integers which turns out to be the class of (partial) recursive functions. The equivalence between the Turing computable functions and the general recursive functions was originally proved via the λ -calculus: the general recursive functions are exactly the λ -definable functions as are the Turing computable functions.

The equivalence between the λ -definable functions and the recursive functions was one of the arguments used by Church to defend his thesis proposing the identification of the intuitive class of effectively computable functions with the class of recursive functions; in fact one can give arguments for the so called Church’s superthesis which states that for the functions involved this identification preserves the intensional character, i.e. process of computation.

Historically the first undecidable problem was constructed by Church as a problem about terms of the λ -calculus (whether they have a normal form). The first definition, due to Church and Kleene, of the recursive ordinals went via the λ -calculus. The fixed point theorem of the λ -calculus inspired Kleene to the recursion theorem. Thus the λ -calculus played a central role in the early investigations of the theory of recursive functions.

Church originally designed the λ -calculus as part of a general system of functions intended to be a foundation of mathematics. The paradox of Kleene and Rosser showed that this system was inconsistent. The present theory was extracted as a consistent subtheory, CHURCH [1941]. After this,

Church seemed to have lost interest in using the λ -calculus to provide a foundation for the whole of mathematics. CURRY et al. [1958, 1972], on the other hand, have developed various systems of illative combinatory logic intended as an ultimate foundation. These systems, however, have not been developed enough to be a satisfactory basis for mathematics. See also SCOTT [1975b] for work in this direction.

Due to the type free approach it was not clear how to construct models of the theory. One would want a set X in which its function space $X \rightarrow X$ can be embedded, contradicting Cantors theorem. This difficulty was overcome by Scott in his D_∞ model constructions (in 1969), by restricting $X \rightarrow X$ to the *continuous* functions on X (provided with a proper topology). Also in the graph model $\mathcal{P}\omega$, continuity plays an essential role.

Scotts models added a new dimension to the theory, namely limit considerations. The author agrees with Scott's claim that this really makes the theory *λ -calculus* and what has been done before should be called *λ -algebra*. See Section 7 where equalities in D_∞ that cannot be proved algebraically are established by approximation methods.

Typed versions of the theory, as well as their connections with category- and proof theory are purposely not considered. The character of the typed theories is totally different from the type free version, e.g. all typed terms have a normal form. See TROELSTRA [1973] as reference for the typed theory of (primitive) recursive functionals and MANN [1975] for the relation with category- and proof theory.

It should be mentioned also that there is a theory related to the λ -calculus, in which application is only partially defined. This is the theory of uniformly reflexive structures of Wagner and Strong. This theory has an obvious model in the partial recursive indices. In fact it is intended to be an axiomatization of parts of recursion theory. See BARENDREGT [1975] for an introduction and references.

Summary

Section 1 gives an introduction to the theory and provides a general model theoretic setting. Section 2 gives a treatment of the classical λ -calculus. It will be proved that the recursive functions can be represented as λ -terms and that many sets of λ -terms are undecidable. In Section 3 the graph model $\mathcal{P}\omega$ is constructed. Section 4 treats Scott's construction of the models D_∞ as a projective limit of complete lattices. In Section 5 a λ -theory \mathcal{H} is introduced. \mathcal{H} has a *unique* maximal consistent extension \mathcal{H}^* . Section 6 associates to each term a tree, useful for the determination of its image in

the models $\mathcal{P}\omega$, D_∞ . In Section 7 it is proved that $D_\infty \models M = N$ iff $M = N \in \mathcal{H}^*$ iff M, N have equivalent trees.

1. Towards the theory

The λ -calculus studies functions and their applicative behavior, and not, as in category theory, just their behavior under composition. Therefore application is the primitive operation of the λ -calculus. The function f applied to the argument a is denoted by fa .

Schönfinkel observed that it is not necessary to introduce functions of more variables. Indeed, for a function of say two variables $f(x, y)$, one can consider g_x with $g_x(y) = f(x, y)$, and then f' with $f'x = g_x$; hence $(f'x)y = f(x, y)$. Therefore a convenient notation is $hx_1 \cdots x_n = (\cdots (hx_1) \cdots x_n)$ (association to the left), the above example becoming $f'xy = f(x, y)$. A similar construction occurs in the s-m-n theorem in recursion theory.

1.1. DEFINITION. An *applicative system* is a structure $\mathfrak{M} = \langle X, \cdot \rangle$, where \cdot is a binary operation (application) on X .

The set of *terms* (using variables a_0, a_1, \dots) over \mathfrak{M} , $T(\mathfrak{M})$, is inductively defined as follows: $x_i \in T(\mathfrak{M})$; $a \in X \Rightarrow c_a \in T(\mathfrak{M})$; $A, B \in T(\mathfrak{M}) \Rightarrow (AB) \in T(\mathfrak{M})$. c_a is the constant corresponding to a . Juxtaposition of terms denotes application.

$A_1 A_2 \cdots A_n$ denotes $(\cdots (A_1 A_2) \cdots A_n)$ (association to the left).

1.2. DEFINITION. A *combinatory algebra* is an applicative system \mathfrak{M} such that \mathfrak{M} is not trivial (i.e. $\text{Card}(X) > 1$) and for each term A over \mathfrak{M} , with variables among y_1, \dots, y_n , we have in \mathfrak{M} :

1.3. $\exists f \forall y_1 \cdots y_n f y_1 \cdots y_n = A$ (*combinatory completeness*).

A combinatory algebra M is *extensional* if in addition in \mathfrak{M}

1.4. $\forall x (fx = f'x) \rightarrow f = f'$ (*extensionality*).

Combinatory completeness expresses that all algebraic functions are representable by an element. The motivation for this axiom is that for functions studied as rules, one certainly would like them to be closed under explicit definition.

It is essential that in 1.3 A is purely algebraic and not defined using logical operations. A diagonalization would otherwise make the system

trivial. However, combinatory completeness is already quite strong, e.g. there are no finite combinatory algebras and in fact no recursive ones. In contrast with e.g. the theory of fields there are 2^{\aleph_0} prime combinatory algebras.

In 1.3 combinatory completeness is expressed by an existential axiom. By an extension of the type of the language this can be expressed in a universal way (cf. the elementary theory of groups where the axiom $\forall x \exists y x \cdot y = e$ can be expressed by $x \cdot x^{-1} = e$ after extending the language with $^{-1}$). In fact there are two ways to do this. The first one, employed by Church, adds to the language an abstraction operator λ : if A is a term, so is $\lambda x.A$. Combinatory completeness now follows from

$$1.5. \quad (\lambda x.A)a = A[x/a] \quad (\beta\text{-conversion}).$$

Multiple abstraction can be replaced by simple ones following Schönfinkel's idea: let $\lambda x_1 \cdots x_n.A = \lambda x_1(\lambda x_2 \cdots (\lambda x_n.A) \cdots)$; then $(\lambda x_1 \cdots x_n.A)a_1 \cdots a_n = A[x_1 \cdots x_n/a_1 \cdots a_n]$.

The other approach, due to Curry, results from realizing that combinatory completeness follows from two of its instances.

1.6. THEOREM. *Let $\mathfrak{M} = \langle X, \cdot \rangle$ be an applicative structure such that for some $k, s \in X$ one has in \mathfrak{M} :*

- (i) $k \neq s$,
- (ii) $kxy = x$,
- (iii) $sxyz = xz(yz)$.

Then \mathfrak{M} is a combinatory algebra.

PROOF. First let $i = skk$; then $ix = skkx = kx(kx) = x$. By induction on the complexity of a term A over \mathfrak{M} one can define $\lambda^*x.A$ and show that in \mathfrak{M} $(\lambda^*x.A)a = A[x/a]$. Let $I \equiv c_i$, $K \equiv c_k$ and $S \equiv c_s$;

$$\lambda^*x.x = I; \quad \lambda^*x.y = Ky \quad \text{if } x, y \text{ are different variables};$$

$$\lambda^*x.c_b = Kc_b; \quad \lambda^*x.A_1A_2 = S(\lambda^*x.A_1)(\lambda^*x.A_2).$$

Therefore for the terms over \mathfrak{M} one can define λ -abstraction satisfying β -conversion, hence \mathfrak{M} is a λ -algebra. \square

Curry's theory is elegant because of its simplicity. In fact the theory of combinators with constants k and s satisfying (i)–(iii) of 1.6 is the simplest theory which is essentially undecidable. Church's notation is more intuitive however and will be used in this chapter.

The (*formal*) λ -calculus is essentially the theory which has application and abstraction as primitives and β -conversion as axiom. In addition it has the notion of *reduction* which formalizes the fact that e.g. an expression like $(\lambda x. x^2 + 1)3$ can be computed to yield 10, but not conversely. Due to the Church–Rosser theorem, reduction is very useful for the proof theory of the λ -calculus.

It should be stressed that in a theory about functions as rules, *terms* play a central role. This view differs with that of Scott, who puts the models central. It is true indeed that models are of interest not only for the insight they give on the equality of terms, but also for their mathematical structure. But the theory of D_∞ is especially beautiful because of the limit characterization of equality of *terms*; see Section 7.

Typical questions asked about terms are:

- (i) What kind of functions on terms are representable?
- (ii) Which terms are equal, which ones essentially different? Which terms can/should be equated?

Restricted to numerals, the classical answer to (i) is: the recursive functions. Question (ii) can be approached by either giving consistency proofs for reasonable extensions of the λ -calculus or by constructing models and considering the set of equations true in them. \mathcal{H} and \mathcal{H}^* of Section 5 were found by the first and second method respectively. Also the questions under (ii) explain why extensionality is sometimes added to the λ -calculus. Cf. the theorem of Böhm in 2.23, which states that the extensional theory is complete with respect to terms having a nf.

The theory

1.7. DEFINITION. The λ -calculus has the following language.

Alphabet:

- a_0, a_1, \dots variables
- $\rightarrow, =$ reduction, equality;
- $\lambda,), ($ auxiliary symbols.

Terms are inductively defined:

- (i) Any variable is a term;
- (ii) if M, N are terms, so is (MN) ;
- (iii) if M is a term and x a variable, then $(\lambda x M)$ is a term.

Formulas:

If M, N are terms, then $M \rightarrow N$ and $M = N$ are formulas.

1.8. CONVENTIONS. A term of the λ -calculus is called a λ -*term*. M, N, \dots is a syntactic notation for arbitrary λ -terms. x, y, z, \dots is a syntactic notation

for arbitrary variables. $M_1 M_2 \cdots M_n$ stands for $(\cdots (M_1 M_2) \cdots M_n)$ (association to the left). $\lambda x_1 \cdots x_n . M$ stands for $(\lambda x_1 (\lambda x_2 \cdots (\lambda x_n M) \cdots))$. The symbol \equiv denotes syntactic equality.

A variable occurs *free* in a term M if x is not in the scope of a λx , otherwise x occurs *bound*. In this respect λx has the same binding properties as $\forall x$ in predicate logic or $\int_b^a \cdots dx$ in calculus. We identify terms differing only in the names of their bound variables, e.g. $\lambda x . x \equiv \lambda y . y$. $FV(M)$ is the set of free variables in M , M is *closed* if $FV(M) = \emptyset$.

$M[x/N]$ denotes the result of substituting N for the *free* occurrences of x in M . In order to prevent confusion of variables we have to assume, as is the case in predicate logic, that no free variable of N becomes bound in $M[x/N]$. This can be accomplished by renaming some of the bound variables in M , e.g. $(\lambda a . ax)[x/a] \equiv \lambda a' . a'a \neq \lambda a . aa$. After this precaution, the definition of substitution is independent of the choice of representative in the equivalence class of identified terms. See also DE BRUIJN [1972].

1.9. DEFINITION. The λ -calculus is defined by the following axiom schemes and rules.

- I. 1. $(\lambda x . M)N \rightarrow M[x/N]$ (β -reduction),
2. $M \rightarrow M$,
3. $M \rightarrow N, N \rightarrow L \Rightarrow M \rightarrow L$,
4. (a) $M \rightarrow M' \Rightarrow ZM \rightarrow ZM'$,
- (b) $M \rightarrow M' \Rightarrow MZ \rightarrow M'Z$,
- (c) $M \rightarrow M' \Rightarrow \lambda x . M \rightarrow \lambda x . M'$.
- II. 1. $M \rightarrow M' \Rightarrow M = M'$,
2. $M = M' \Rightarrow M' = M$,
3. $M = N, N = L \Rightarrow M = L$,
4. (a) $M = M' \Rightarrow ZM = ZM'$,
- (b) $M = M' \Rightarrow MZ = M'Z$,
- (c) $M = M' \Rightarrow \lambda x . M = \lambda x . M'$.

If $M = N$ or $M \rightarrow N$ is derivable one writes $\lambda \vdash M = N$ or $(\lambda \vdash) M \rightarrow N$ respectively. Since $=$ is generated by \rightarrow , the rules II.4 follow from I.4. The addition of II.4 is necessary however, if one considers extensions of the λ -calculus.

1.10. Extensionality

The λ -calculus can be extended by the following rule of extensionality,

ext: $Mx = M'x \Rightarrow M = M'$, provided $x \notin FV(M), FV(M')$.

The rule *ext* can be axiomatized by adding the rule

$$\lambda x. Mx \rightarrow M \quad (\eta\text{-reduction})$$

provided $x \notin \text{FV}(M)$:

If $Mx = M'x$, and $x \notin \text{FV}(MM')$, then $\lambda x. MX = \lambda x. M'x$, hence $M = \lambda x. Mx = \lambda x. M'x = M'$.

The λ -calculus with this additional reduction rule is called the $\lambda\eta$ -calculus. Provability in this theory is denoted by $\lambda\eta \vdash \dots$.

M and N are $\beta(\eta)$ -convertible iff $\lambda(\eta) \vdash M = N$.

The models

Our main object of study is the λ -calculus. Therefore we would want that the combinatory algebras are also models for this theory, i.e. that there is an interpretation of the λ -operator. However unless a combinatory algebra is extensional, there is a choice for the element f representing the function A in 1.3. Thus the combinatory algebras have not enough structure to be models for the λ -calculus.

1.11. DEFINITION. A *pre- λ -algebra* is a combinatory algebra \mathfrak{M} together with a method of assigning to each term $A \in T(\mathfrak{M})$ a term $\lambda^*x. A \in T(\mathfrak{M})$ such that

- (i) x does not occur in $\lambda^*x. A$,
- (ii) $\mathfrak{M} \models (\lambda^*x. A)x = A$.

Remarks. (1) For most \mathfrak{M} this assignment $A \mapsto \lambda^*x. A$ is provided for by the *proof* that \mathfrak{M} is a combinatory algebra.

(2) Although the definition of a pre- λ -algebra is not formulated in a conventional first order way, from a constructive point of view it is completely clear.

1.12. DEFINITION. *Interpretation of λ -terms.* Let ρ be a valuation of the variables into a pre- λ -algebra $\mathfrak{M} = \langle X, \cdot \rangle$ i.e. $\rho = \{a_0, a_1, \dots\} \rightarrow X$. The value in \mathfrak{M} of a λ -term M under the valuation ρ , notation $\llbracket M \rrbracket_\rho^{\mathfrak{M}}$, is defined in two steps. First M is transformed into a term $\langle\langle M \rangle\rangle^{\mathfrak{M}} \in T(\mathfrak{M})$. Then this term is interpreted in M under the valuation ρ in the usual first order way, yielding $\llbracket M \rrbracket_\rho^{\mathfrak{M}}$.

$\langle\langle M \rangle\rangle^{\mathfrak{M}}$ is inductively defined as follows: $\langle\langle x \rangle\rangle^{\mathfrak{M}} = x$; $\langle\langle MN \rangle\rangle^{\mathfrak{M}} = \langle\langle M \rangle\rangle^{\mathfrak{M}} \langle\langle N \rangle\rangle^{\mathfrak{M}}$; $\langle\langle \lambda x. M \rangle\rangle^{\mathfrak{M}} = \lambda^*x. \langle\langle M \rangle\rangle^{\mathfrak{M}}$.

1.13. DEFINITION. *Satisfaction.* As usual, $\mathfrak{M} \models M = N$ iff for all ρ , $\llbracket M \rrbracket_\rho^{\mathfrak{M}} = \llbracket N \rrbracket_\rho^{\mathfrak{M}}$.

1.14. DEFINITION. A λ -algebra or model of the λ -calculus is a pre- λ -algebra \mathfrak{M} such that $\lambda \vdash M = N \Rightarrow \mathfrak{M} \models M = N$.

Remark. The term model $\mathfrak{M}(\text{CL})$ of the theory of combinators is a pre- λ -algebra (by the proof of 1.6) but not a λ -algebra, since $\mathfrak{M}(\text{CL}) \not\models \lambda x.((\lambda y.y)x) = \lambda x.x : s(ki)i \neq i$.

1.15. DEFINITION. (i) A weakly extensional (w.e.) λ -algebra is a λ -algebra \mathfrak{M} such that

$$\mathfrak{M} \models M = M' \Rightarrow \mathfrak{M} \models \lambda x.M = \lambda x.M'.$$

(ii) A λ -algebra \mathfrak{M} is *extensional* iff \mathfrak{M} satisfies $\forall x (fx = gx) \rightarrow f = g$ (extensionality).

1.16. REMARK. (1) An extensional λ -algebra is clearly w.e.

(2) A combinatory algebra satisfying extensionality is an extensional λ -algebra, since there is only one way to define abstraction.

(3) There are interesting λ -algebras that are not weakly extensional, e.g. $\mathfrak{M}^0(\lambda)$ (see below) as follows by the ω -incompleteness of the λ -calculus (cf. PLOTKIN [1974]) or \mathcal{P}_ω^0 .

(4) The only λ -algebras that are considered here are either w.e. or term models.

1.17. General concepts and notations

ω denotes the set of natural numbers. $\lambda x. \dots$ denotes the mapping $x \mapsto \dots$ (meta lambda).

Notions connected with theories

Λ is the set of λ -terms. Λ^0 is the set of closed λ -terms. Let \mathbf{T} be a set of equations between λ -terms. Then $\lambda + \mathbf{T}$ is the λ -calculus extended with the equations in \mathbf{T} as axioms. $\mathbf{T}^+ = \{M = N \mid M, N \in \Lambda^0 \text{ and } \lambda + \mathbf{T} \vdash M =$

$N\}$. \mathbf{T} is said to be *consistent* if \mathbf{T}^+ does not contain every equation. A λ -theory is a consistent set of equations \mathbf{T} such that $\mathbf{T} = \mathbf{T}^+$.

λ is the λ -theory $\{M = N \mid M, N \in \Lambda^0 \text{ and } \lambda \vdash M = N\}$; the consistency is shown in 2.9.

For a λ -theory \mathbf{T} define $M \sim_{\mathbf{T}} N$ iff $\lambda + \mathbf{T} \vdash M = N$. $\sim_{\mathbf{T}}$ is an equivalence relation and let $[M]_{\sim_{\mathbf{T}}}$ denote the equivalence class of M with respect to $\sim_{\mathbf{T}}$. The *term model* of \mathbf{T} , $\mathfrak{M}(\mathbf{T})$, is the λ -algebra consisting of all λ -terms modulo $\sim_{\mathbf{T}}$, with application and abstraction defined canonically. The *closed term model* of \mathbf{T} , $\mathfrak{M}^0(\mathbf{T})$, is the set of terms without free variables modulo $\sim_{\mathbf{T}}$.

A λ -theory \mathbf{T} is *maximally consistent* if \mathbf{T} has no proper consistent extensions.

Notions connected with models

For a λ -algebra \mathfrak{M} , $\text{Th}(\mathfrak{M})$ is the λ -theory $\{M = N \mid M, N \in \Lambda^0 \text{ and } \mathfrak{M} \models M = N\}$. The consistency follows from the fact that $\text{Card}(\mathfrak{M}) > 1$ (see 2.3).

The *interior* of \mathfrak{M} , notation \mathfrak{M}^0 , is the substructure consisting of the images in \mathfrak{M} of the closed λ -terms. Up to isomorphism $\mathfrak{M}^0 = \mathfrak{M}^0(\text{Th}(\mathfrak{M}))$. \mathfrak{M} is *hard* iff $\mathfrak{M} = \mathfrak{M}^0$. The hard λ -algebras are the prime structures among the λ -algebras.

For λ -algebras a homomorphism $h : \mathfrak{M} \rightarrow \mathfrak{M}'$ should not only preserve application, but also abstraction, i.e. for a term $A \in T(\mathfrak{M})$, $h(\lambda^*x.A) = \lambda^*x.hA$ in \mathfrak{M}' where for $B \in T(\mathfrak{M})$, $hB \in T(\mathfrak{M}')$ is the term obtained by replacing in B all constants c_a by c_{ha} .

We will use homomorphisms only in connection with term models. There the description is simple. If $\mathbf{S} \subset \mathbf{T}$ are λ -theories, then a $h : \mathfrak{M}(\mathbf{S}) \rightarrow \mathfrak{M}(\mathbf{T})$ is defined by $h([M]_{\sim_{\mathbf{S}}}) = [M]_{\sim_{\mathbf{T}}}$. Thus each (closed) term model $\mathfrak{M}^{(0)}(\mathbf{S})$ is the homomorphic image of $\mathfrak{M}^{(0)}(\lambda) : [M]_{\sim_{\lambda}} \rightarrow [M]_{\sim_{\mathbf{S}}}$. If \mathbf{S} is maximally consistent, then $\mathfrak{M}^{(0)}(\mathbf{S})$ is algebraically *simple*, i.e. has no proper homomorphic images.

2. Classical λ -calculus

The classical theory is mainly concerned with $\mathfrak{M}(\lambda)$. Among others the following theorems will be proved. All recursive functions are λ -definable (Kleene). The set of terms with(out) a normal form is undecidable (Church). There is no recursive model for the λ -calculus (Grzegorzcyk). The last two theorems follows most easily from a theorem of Scott. Finally

the theorem of Böhm is stated, which shows the completeness of $\beta\eta$ -conversion for terms having a normal form.

2.1. FIXED POINT THEOREM. *For every $F \in \Lambda$ there is an $M \in \Lambda$ such that $\lambda \vdash FM = M$.*

PROOF. Define $\omega = \lambda x. F(xx)$ and $M = \omega\omega$. Then $\lambda \vdash M = \omega\omega = (\lambda x. F(xx))\omega = F(\omega\omega) = FM$. \square

Remarks. (1) The fixed points can be found in a uniform way: let $Y = \lambda f. (\lambda x. f(xx))(\lambda x. f(xx))$; then $\lambda \vdash Yf = f(Yf)$.

(2) Since the theorem holds for terms possibly containing free variables, each element of a λ -algebra has a fixed point.

(3) Curry calls Y the *paradoxical combinator*.

Note that in 2.1, $\lambda \vdash M \rightarrow FM$. This explains why the related constructions in the recursion theorem or Gödel's self-referential sentence are somewhat puzzling, cf. BARENDREGT [to appear], §6.7.

2.2. Frequently we need some standard terms. Let $I = \lambda x. x$, $K = \lambda xy. x$, $S = \lambda xyz. xz(yz)$ and $\Omega = (\lambda x. xx)(\lambda x. xx)$. Then $\lambda \vdash IM = M$, $\lambda \vdash KMM = M$ and $\lambda \vdash SMNL = ML(NL)$.

From 1.6 it follows that each closed term can be defined in terms of I , K and S .

2.3. Truth values **t** (true) and **f** (false). Define **t** = K , **f** = KI . Then $\lambda \vdash \mathbf{t}MN = M$ and $\lambda \vdash \mathbf{f}MN = N$. Note that $\mathbf{t} \neq \mathbf{f}$ in any λ -algebra \mathfrak{M} , for otherwise \mathfrak{M} would satisfy $x = \mathbf{t}xy = \mathbf{f}xy = y$ and hence be trivial.

2.4. Conditional

If B is a term taking values **t** and **f**, then the intuitive value of “If B then M else N ” can be represented by BMN .

2.5. Ordered pairs

Define $[M, N] = \lambda x. xMN$, $(M)_0 = M\mathbf{t}$ and $(M)_1 = M\mathbf{f}$. Then $\lambda \vdash ([M_0, M_1])_i = M_i$, $i = 0, 1$.

2.6. Numerals

Define $\underline{0} = I$ and $\underline{n+1} = [\underline{n}, K]$. The numerals are chosen this way to

provide a convenient base for the representation of the recursive functions. CHURCH [1941] used the following numerals: $\underline{n} = \lambda fx. f^n x$, where $f^0 x = x$ and $f^{n+1} x = f(f^n x)$.

2.7. DEFINITION. (i) A *redex* is a term of the form $(\lambda x. P)Q$ (in the extensional theory also $(\lambda x. Px)$, with $x \notin FV(P)$, is a redex).

(ii) A term M is *in normal form* (nf) iff there is no subterm of M which is a redex (if it is necessary to distinguish nf's in the λ - and $\lambda\eta$ -calculus one talks about β - and $\beta\eta$ -nf's).

(iii) A term M *has a nf* iff for some N in nf $\lambda \vdash M \rightarrow N$ (in the extensional theory $\lambda\eta \vdash M \rightarrow N$).

Intuitively a term is in nf if it cannot be computed any further.

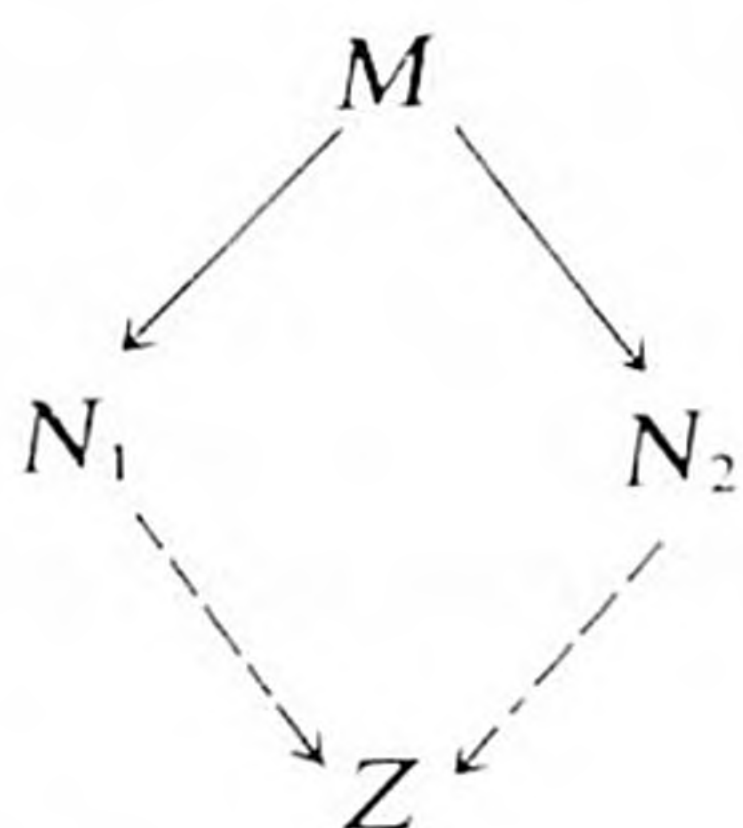
Example. $(\lambda x. xx)y$ has the nf yy ; Ω has no nf. Note that for every natural number n , \underline{n} is in nf.

Now an important theorem on reduction will be stated. For details see e.g. HINDLEY et al. [1972] p. 139 or BARENDREGT [to appear].

2.8. CHURCH-ROSSER THEOREM. *If $\lambda \vdash M = N$, then for some Z , $\lambda \vdash M \rightarrow Z$ and $\lambda \vdash N \rightarrow Z$ (and similarly for the $\lambda\eta$ -calculus).*

PROOF (outline; after Tait and Martin-Löf). The theorem follows from (and in fact is equivalent with):

(*) If $M \rightarrow N_1$, $M \rightarrow N_2$, then for some Z , $N_1 \rightarrow Z$ and $N_2 \rightarrow Z$:



diamond property for \rightarrow .

That 2.8 follows from (*) is proved by induction on the length of proof of $\lambda \vdash M = N$, (*) being needed for the transitivity of $=$.

(*) is proved by defining a relation \rightarrow^o on λ -terms, such that (i) \rightarrow^o has the diamond property; (ii) \rightarrow is the transitive closure of \rightarrow^o . The diamond property for \rightarrow then follows by a simple diagram chasing.

The relation \rightarrow^o is defined inductively as follows:

$$M \rightarrow M: \quad M \rightarrow M', N \rightarrow N' \Rightarrow (\lambda x. M)N \rightarrow M'[x/N'];$$

$$M \rightarrow M', N \rightarrow N' \Rightarrow MN \rightarrow M'N': \quad M \rightarrow M' \Rightarrow \lambda x. M \rightarrow \lambda x. M'.$$

Then (ii) is obvious and (i) follows from a case analysis and $M \rightarrow M'$, $N \rightarrow N' \Rightarrow M[x/N] \rightarrow M'[x/N']$, as can be proved by induction on the generation of \rightarrow . \square

- 2.9. COROLLARY.** (i) *If M has a nf at all, then M has a unique nf.*
(ii) *If $\lambda \vdash M = N$ and N is in nf, then $M \rightarrow N$.*
(iii) *The λ -calculus is consistent, i.e. $\lambda \not\vdash M = N$ for some equation.*

PROOF. (i) Note that if N is a nf and $N \rightarrow N'$, then $N' \equiv N$. Now suppose $M \rightarrow N_1$, $M \rightarrow N_2$ and N_1, N_2 are nf, then $\lambda \vdash N_1 = N_2$, so by 2.8, $N_1 \rightarrow Z$ and $N_2 \rightarrow Z$. But then $N_1 \equiv Z \equiv N_2$.

(ii) If $M = N$, again $M \rightarrow Z$ and $N \rightarrow Z$. But since N is a nf, $N \equiv Z$, so $M \rightarrow N$.

(iii) If M, N are distinct nf's, then $\lambda \not\vdash M = N$ by (ii) and (i). \square

2.10. DEFINITION. Let ω be the set of natural numbers. A function $f: \omega^n \rightarrow \omega$ is λ -definable iff for some $F \in \Lambda^0$,

$$(*) \quad \lambda \vdash F\underline{k}_1 \cdots \underline{k}_n = \underline{m} \Leftrightarrow f(k_1, \dots, k_n) = m.$$

If $(*)$ holds, then f is said to be λ -definable by F .

2.11. REMARK. If for some λ -term F instead of $(*)$ we have

$$(**) \quad f(k_1, \dots, k_n) = m \Rightarrow \lambda \vdash F\underline{k}_1 \cdots \underline{k}_n = \underline{m},$$

then f is λ -definable by F : suppose $\lambda \vdash F\underline{k}_1 \cdots \underline{k}_n = \underline{m}$ and $f(k_1, \dots, k_n) = m'$. Then by $(**)$, $\lambda \vdash F\underline{k}_1 \cdots \underline{k}_n = \underline{m}'$ and hence by 2.9(i) $m = m'$, since the numerals are in normal form. So $f(k_1, \dots, k_n) = m$.

2.12. LEMMA. *The initial functions $U_i^n(x_1, \dots, x_n) = x_i$, $Z(x) = 0$, $S^+(x) = x + 1$ are λ -definable.*

PROOF. Take as defining terms $U_i^n = \lambda x_1 \cdots x_n. x_i$, $Z = \lambda x. \underline{0}$ and $S^+ = \lambda x. [x, K]$ respectively and use 2.11. \square

2.13. LEMMA. *The λ -definable functions are closed under composition.*

PROOF. The representation of the composition is the composition of the representations. \square

2.14. LEMMA. *There are terms P and Zero , such that $P(S^+x) = x$ and $\text{Zero } x = \mathbf{t}$ if $x = \underline{0}$, $\text{Zero } x = \mathbf{f}$ if x is a numeral $\neq \underline{0}$.*

PROOF. Take $P = \lambda x.(x)_0$, $\text{Zero} = \lambda x.(x)_1 \mathbf{f} \mathbf{t}$. \square

2.15. LEMMA. *The λ -definable functions are closed under primitive recursion.*

PROOF. For simplicity we omit parameters. Let f be defined by $f(0) = k$, $f(n+1) = g(f(n), n)$, where g is λ -definable by G . We want to define F such that it satisfies $Fx = \text{if } \text{Zero } x, \text{ then } \underline{k}, \text{ else } G(F(Px))(Px)$. By 2.4 this can be expressed as $Fx = \text{Zero } x \underline{k} [G(F(Px))(Px)]$ or $F = \lambda x. \text{Zero } x \underline{k} [G(F(Px))(Px)]$. Define $\Theta = \lambda fx. \text{Zero } x \underline{k} [G(f(Px))(Px)]$. Then we can take F as being the fixed point of Θ . \square

2.16. LEMMA. *The λ -definable functions are closed under minimalization.*

PROOF. (Again we omit parameters.) Let f be defined by $f(x) = \mu y [g(x, y) = 0]$, where g is λ -defined by G and $\forall n \exists m g(n, m) = 0$. As in 2.15 we can find a λ -term H such that $Hxy = \text{if } \text{Zero}(Gxy), \text{ then } y, \text{ else } Hx(S^+y)$. Then take $F = \lambda x. Hx \underline{0}$. \square

Remark. It is clear that the λ -calculus is a recursively axiomatizable theory. Hence (after Gödelization) the relation $\{(M, N) \mid \lambda \vdash M = N\}$ is recursively enumerable.

2.17. THEOREM (Kleene). *The λ -definable functions are exactly the recursive functions.*

PROOF. If f is recursive, then f is λ -definable, since the recursive functions are the least class containing the initial functions which is closed under composition, primitive recursion and minimalization. If f is λ -defined by F , then $f(k_1, \dots, k_n) = m \Leftrightarrow \lambda \vdash F \underline{k}_1 \cdots \underline{k}_n = \underline{m}$ hence by the preceding remark the graph of f is r.e., so f is recursive. \square

Remark. A partial function $\psi : \omega^k \rightarrow \omega$ is λ -definable iff for some term F , $\psi(\vec{k}) = m \Leftrightarrow \lambda \vdash F \vec{k} = \underline{m}$, and $\psi(\vec{k})$ undefined $\Leftrightarrow F \vec{k}$ has no nf.

It can be shown that for partial functions, ψ is λ -definable iff ψ is partial recursive. In order to do this one must show that certain terms have no nf. For this purpose the following reduction strategy is useful.

2.18. DEFINITION. Let M be a λ -term. If M is not a nf, let M' be obtained by reducing the leftmost redex in M (i.e. the redex with its λ as left as possible), else M' is undefined. Define $M_0 = M$, $M_{n+1} = (M_n)'$. The sequence $M_0 \rightarrow M_1 \rightarrow \dots$ (finite or infinite) is called the *leftmost* or *normal* reduction chain of M .

2.19. NORMALIZATION THEOREM (Curry). M has a nf iff the leftmost reduction chain of M is finite.

PROOF. See CURRY et al. [1958] p.142. \square

The theorem is false for arbitrary reduction chains, consider e.g. $KI\Omega$ having a nf but also an infinite reduction chain.

Undecidability results

After coding syntactical objects as natural numbers one can speak of the decidability of a set of terms or equations. It will be shown that the λ -calculus is essentially undecidable, i.e. has no decidable consistent extension.

It is standard to define a coding $M \rightarrow \#M$ such that there are recursive functions $Ap(\#M, \#N) = \#(MN)$ and $Num(n) = \#\underline{n}$. The numeral $\underline{\#M}$ will be denoted by $\ulcorner M \urcorner$.

2.20. SECOND FIXED POINT THEOREM. For each $F \in \Lambda^{(0)}$ there is an $X \in \Lambda^{(0)}$ such that $\lambda \vdash F \ulcorner X \urcorner = X$.

PROOF. Let the recursive functions Ap and Num be λ -defined by the terms Ap and Num . Define $\omega = \lambda x. F(Ap x (Num x))$ and $X = \omega \ulcorner \omega \urcorner$. Then

$$\lambda \vdash X = \omega \ulcorner \omega \urcorner \rightarrow F(Ap \ulcorner \omega \urcorner (Num \ulcorner \omega \urcorner)) \rightarrow F \ulcorner \omega \urcorner \ulcorner \omega \urcorner = F \ulcorner X \urcorner,$$

since $Num \ulcorner \omega \urcorner \rightarrow \ulcorner \omega \urcorner$. \square

A set $\mathcal{A} \subseteq \Lambda^0$ is closed under equality iff $M \in \mathcal{A}$ and $M = N \in \Lambda \Rightarrow N \in \mathcal{A}$. $\mathcal{A} \subseteq \Lambda^0$ is non-trivial iff $\mathcal{A} \neq \emptyset$ and $\mathcal{A} \neq \Lambda^0$.

2.21. THEOREM (Scott). Let $\mathcal{A} \subseteq \Lambda^0$ be a non-trivial set closed under equality. Then \mathcal{A} is not recursive.

PROOF. Let $M_0 \in \mathcal{A}$, $M_1 \in \Lambda^0 - \mathcal{A}$. Suppose \mathcal{A} were recursive. Then there is a recursive function $f : \omega \rightarrow \{0, 1\}$ such that $f(\#M) = 0$ iff $M \in \mathcal{A}$. Let f be λ -defined by F . Define $F' = \lambda x. \text{If Zero}(Fx), \text{ then } M_1 \text{ else, } M_0$. Then $F'(\ulcorner M \urcorner) = M_1$ if $M \in \mathcal{A}$ and $F'(\ulcorner M \urcorner) = M_0$ if $M \notin \mathcal{A}$. $\exists X \in \Lambda^0 F'(\ulcorner X \urcorner) = X$, by 2.20. If $X \in \mathcal{A}$, then $X = F'(\ulcorner X \urcorner) = M_1 \notin \mathcal{A}$ and if $X \notin \mathcal{A}$ then $X = F'(\ulcorner X \urcorner) = M_0 \in \mathcal{A}$, contradicting in both cases that \mathcal{A} is closed under equality. \square

2.22. COROLLARY (Church; Grzegorzcyk). (i) *The set of terms with(out) a nf is not recursive.*

(ii) *There are no recursive λ -theories.*

(iii) *There are no recursive λ -algebras.*

PROOF. (i) $\{M \mid M \text{ has a (has no) nf}\}$ satisfies the condition of 2.21.

(ii) Let \mathbf{T} be a λ -theory. Then $\{M \mid M = I \in \mathbf{T}\}$ satisfies 2.21. Hence \mathbf{T} is not recursive.

(iii) If \mathcal{M} were a recursive λ -algebra, then $\text{Th}(\mathcal{M})$ would be a recursive λ -theory, contradicting (ii). \square

The following result shows the completeness of the $\lambda\eta$ -calculus with respect to terms having a nf.

2.23. THEOREM (Böhm). *If M, N are different $\beta\eta$ -nf's, then $\lambda + M = N$ is inconsistent.*

PROOF (outline). If $M \neq N$ are $\beta\eta$ -nf's, then M, N have finite Böhm trees which are not η -equivalent, see Section 6. Hence by 6.8, $M = N \notin \mathcal{H}^*$. In fact it follows from the proof of 6.8 that in this case $\lambda \vdash C[M] = x$ and $\lambda \vdash C[N] = y$ for some context $C[\cdot]$ and variables $x \neq y$. So $\lambda + M = N \vdash x = y$, i.e. $\lambda + M = N$ is inconsistent. \square

For terms without nf this completeness result is false, see 7.2.

3. Construction of the graph model $\mathcal{P}\omega$

The first λ -algebras were the lattice models D_∞ constructed by Scott in 1969, see Section 4. The graph model $\mathcal{P}\omega$ is less involved and therefore will be described first. This model was found by PLOTKIN [1972] and, in a more

explicit form, by Scott. For connections of this model with recursion theoretic ideas, see SCOTT [1975a], [1976].

The role of continuity in the model construction was already mentioned in Section 1. The topology on $\mathcal{P}\omega$ is such that a continuous function $\mathcal{P}\omega \rightarrow \mathcal{P}\omega$ can be coded as an element of $\mathcal{P}\omega$. This is the essential feature of the model.

3.1. DEFINITION. (i) $\mathcal{P}\omega = \{x \mid x \subseteq \omega\}$.

(ii) There are countably many finite elements of $\mathcal{P}\omega$. As an effective one-one enumeration of these sets we use e_n , where

$$e_n = \{k_0, \dots, k_{m-1}\} \quad \text{with } k_0 < \dots < k_{m-1} \Leftrightarrow n = \sum_{i < m} 2^{k_i}.$$

(iii) (\cdot, \cdot) is the coding of pairs of integers into the integers defined by $(n, m) = \frac{1}{2}(n+m)(n+m+1) + m$.

3.2. DEFINITION. Let $e \subseteq \omega$ be a finite subset.

$$O_e = \{x \in \mathcal{P}\omega \mid e \subseteq x\}; \quad O_n = O_{e_n}.$$

3.3. LEMMA. The $\{O_n\}_{n \in \omega}$ form a base for a topology on $\mathcal{P}\omega$.

PROOF. $O_e \cap O_{e'} = O_{e \cup e'}$. \square

Henceforth we always will consider $\mathcal{P}\omega$ provided with this topology.

3.4. LEMMA. $f: \mathcal{P}\omega \rightarrow \mathcal{P}\omega$ is continuous $\Leftrightarrow f(x) = \bigcup \{f(e_n) \mid e_n \subseteq x\}$.

PROOF. (\Rightarrow) First note that f is monotonic. For suppose $x \subseteq y$ and $n \in f(x)$. Then $f(x) \in O_{\{n\}}$. By continuity for some e , $x \in O_e$ and $f(O_e) \subseteq O_{\{n\}}$. Now $e \subseteq x \subseteq y$, so $y \in O_e$, hence $f(y) \subseteq O_{\{n\}}$, i.e. $n \in f(y)$. Therefore indeed $x \subseteq y \Rightarrow f(x) \subseteq f(y)$.

By monotonicity of f , $f(x) \supseteq \bigcup \{f(e_n) \mid e_n \subseteq x\}$. To show the reverse, suppose $n \in f(x)$. Again for some e , $x \in O_e$ and $f(O_e) \subseteq O_{\{n\}}$. Hence since $e \in O_e$, we have $f(e) \in O_{\{n\}}$, i.e. $n \in f(e) \subseteq \bigcup \{f(e_n) \mid e_n \subseteq x\}$.

(\Leftarrow) Again f is monotonic. For suppose $x \subseteq y$ and $n \in f(x)$. By the assumption, $n \in f(e)$ for some finite $e \subseteq x$. Hence $n \in \bigcup \{f(e') \mid e' \subseteq y\} = f(y)$.

Now suppose $f(x) \in O_e$, i.e. $e \subseteq f(x)$. Let $e = \{m_1, \dots, m_k\}$. By assumption $m_i \in f(e_{n_i})$ for some $e_{n_i} \subseteq x$. Then $f(O_{e_{n_i}}) \subseteq O_{\{m_i\}}$ by monotonicity.

Then $O = O_{e_{n_1}} \cap \cdots \cap O_{e_{n_k}}$ is a neighborhood of x and

$$f(O) \subseteq f(O_{e_{n_1}}) \cap \cdots \cap f(O_{e_{n_k}}) \subseteq O_{\{m_1\}} \cap \cdots \cap O_{\{m_k\}} = O_{\{m_1, \dots, m_k\}} = O_e.$$

So $\forall O_e \ni f(x) \exists x \ni O f(O) \subseteq O_e$, i.e. f is continuous. \square

By the previous lemma, a continuous function f is completely determined by its values on the finite sets. Hence if one knows for what m, n $m \in f(e_n)$, then the $f(e_n)$ are known and therefore f . Thus the information of a continuous function can be coded into a set. This operation is called graph. Its inverse operation is fun.

3.5. DEFINITION. (i) Let $f: \mathcal{P}\omega \rightarrow \mathcal{P}\omega$ be continuous, then $\text{graph}(f) = \{(n, m) \mid m \in f(e_n)\}$.

(ii) Let $u \in \mathcal{P}\omega$. The function $\text{fun}(u)$ is defined by $\text{fun}(u)(x) = \{m \mid \exists e_n \subseteq x (n, m) \in u\}$.

3.6. THEOREM. (i) A continuous function f is uniquely determined by its graph: $\text{fun}(\text{graph}(f)) = f$.

(ii) For every $u \in \mathcal{P}\omega$, $\text{fun}(u)$ is continuous.

PROOF. (i)

$$\begin{aligned} \text{fun}(\text{graph}(f))(x) &= \{m \mid \exists e_n \subseteq x (n, m) \in \text{graph}(f)\} \\ &= \{m \mid \exists e_n \subseteq x m \in f(e_n)\} \\ &= \{f(e_n) \mid e_n \subseteq x\} = f(x), \quad \text{by 3.4.} \end{aligned}$$

(ii) Let $f = \text{fun}(u)$. Then $f(x) = \bigcup \{u_n \mid e_n \subseteq x\}$, where $u_n = \{m \mid (n, m) \in u\}$. Now

$$\begin{aligned} m \in f(x) &\Leftrightarrow \exists n [e_n \subseteq x \wedge m \in u_n] \Leftrightarrow \exists n [(e_n \subseteq e_n \wedge m \in u_n) \wedge e_n \subseteq x] \\ &\Leftrightarrow \exists n [e_n \subseteq x \wedge m \in f(e_n)] \Leftrightarrow m \in \bigcup \{f(e_n) \mid e_n \subseteq x\}. \end{aligned}$$

So $f(x) = \bigcup \{f(e_n) \mid e_n \subseteq x\}$ and 3.4 applies. \square

In general $\text{graph}(\text{fun}(u)) = u$ does not hold, only \supseteq .

3.7. DEFINITION. Application in $\mathcal{P}\omega$ is defined by $u \cdot x = \text{fun}(u)(x)$.

3.8. LEMMA. A function $f: \mathcal{P}\omega^k \rightarrow \mathcal{P}\omega$ is continuous iff f is continuous in each of its variables separately (i.e. $\lambda x. f(x, y_0)$ and $\lambda y. f(x_0, y)$ are continuous for all x_0, y_0).

PROOF. (\Rightarrow) As ever.

(\Leftarrow) It is sufficient to prove this for $k = 2$. Let $f(x, y)$ be continuous in x and y separately. Then

$$f(x, y) = \bigcup \{f(e_n, y) \mid e_n \subseteq x\} = \bigcup \{f(e_n, e_m) \mid e_n \subseteq x, e_m \subseteq y\}.$$

As in the proof of 3.4 \Leftarrow , it now follows that f is continuous in the product topology sense. \square

3.9. LEMMA (Continuity of application). *Define $\text{Ap} : \mathcal{P}\omega^2 \rightarrow \mathcal{P}\omega$ by $\text{Ap}(u, x) = u \cdot x$; then Ap is continuous.*

PROOF. $\lambda x. \text{Ap}(u, x) = \lambda x (u \cdot x) = \text{fun}(u)$ which is continuous by 3.6(ii).

$$\lambda u. \text{Ap}(u, x) = \lambda u. (u \cdot x) = \lambda u \{m \mid \exists e_n \subseteq x (n, m) \in u\}$$

which is clearly continuous. Now the result follows from 3.8. \square

3.10. LEMMA (Continuity of abstraction). *Let $f(x, \vec{y}) : \mathcal{P}\omega^{k+1} \rightarrow \mathcal{P}\omega$ be continuous. Define $g(\vec{y}) = \text{graph}(\lambda x. f(x, \vec{y}))$. Then $g : \mathcal{P}\omega^k \rightarrow \mathcal{P}\omega$ is continuous.*

PROOF. For simplicity we set $k = 1$. Note that by 3.8, g is well defined. Now

$$\begin{aligned} g(y) &= \text{graph}(\lambda x. f(x, y)) = \{(n, m) \mid m \in f(e_n, y)\} \\ &= \{(n, m) \mid m \in \bigcup \{f(e_n, e) \mid e \subseteq y\}\} = \{(n, m) \mid \exists e \subseteq y m \in f(e_n, e)\} \\ &= \bigcup \{(n, m) \mid m \in f(e_n, e)\} \mid e \subseteq y\} = \bigcup \{g(e) \mid e \subseteq y\}. \end{aligned}$$

Hence 3.4 applies. \square

3.11. THEOREM. $\langle \mathcal{P}\omega, \cdot \rangle$ is a w.e. λ -algebra.

PROOF. For a continuous $f : \mathcal{P}\omega^{k+1} \rightarrow \mathcal{P}\omega$, define $\lambda^* d. f(d, \vec{e}) = \text{graph}(\lambda d. f(d, \vec{e}))$. By 3.10 this is a continuous function in \vec{e} and by 3.7 and 3.6(i) one has $(\lambda^* d. f(d, \vec{e})) \cdot d = f(d, \vec{e})$. Now for $A \equiv A(x, y_1, \dots, y_k) \in T(\mathcal{P}\omega)$, where $\{\vec{y}\} = \text{FV}(A) - \{x\}$, define $\lambda^* x. A = c_a \cdot \vec{y}$, where $a = \lambda^* e_1 \cdots \lambda^* e_k \lambda^* d. (A(c_d, c_{e_1}, \dots, c_{e_k})^{\mathcal{P}\omega})$ and $\cdots^{\mathcal{P}\omega}$ denotes the

interpretation in $\mathcal{P}\omega$. This makes $\mathcal{P}\omega$ into a λ -algebra. Since λ^* is defined by functions in extension, $\mathcal{P}\omega$ with λ^* is w.e. \square

SCOTT [1975a] considers an extension of the lambda calculus, called LAMBDA, together with an interpretation in $\mathcal{P}\omega$. It is proved that the interior of $\mathcal{P}\omega$ with respect to LAMBDA consists exactly of the recursively enumerable sets.

4. Construction of D_∞

The results in this section are due to SCOTT [1972]. Again continuity is the essential feature in the model construction.

First complete lattices and their induced topologies are considered. Then follows the construction of the λ -algebras D_∞ as a projective (and at the same time direct) limit of these lattices.

4.1. DEFINITION. Let D be a complete lattice, i.e. a partially ordered set $\langle D, \sqsubseteq \rangle$ such that each subset $X \subset D$ has a supremum $\bigsqcup X \in D$. Then each subset X has an infimum as well:

$$\bigsqcap X = \bigsqcup \{z \mid z \sqsubseteq X\} \quad \text{where } z \sqsubseteq X \Leftrightarrow \forall x \in X \, z \sqsubseteq x.$$

Top, $\top = \bigsqcup D$, and bottom $\perp = \bigsqcap D$ are resp. the largest and smallest elements of D . A subset $X \subset D$ is *directed* iff $\forall x, y \in X \, \exists z \in X \, x, y \sqsubseteq z$. Further, $x \sqcup y$ ($x \sqcap y$) is the supremum (infimum) of $\{x, y\}$. D, D', D'', \dots will range over complete lattices.

4.2. DEFINITION. A subset $U \subset D$ is *open* iff

- (i) $x \in U$ and $x \sqsubseteq y \Rightarrow y \in U$, and
- (ii) $\bigsqcup X \in U \Rightarrow X \cap U \neq \emptyset$ for all directed $X \subset D$.

D and \emptyset are open, and open sets are closed under arbitrary unions; if U_1, U_2 are open, then $U_1 \cap U_2$ is open by the fact that in (ii), X is directed. Hence the partial ordering induces a topology on D . Note that the sets $U_x = \{z \mid z \not\sqsubseteq x\}$ are open and $x \notin U_x$. Therefore the topology is T_0 : if x, y are different, say $x \not\sqsubseteq y$, then $x \in U_y$, $y \notin U_y$. The space is not T_1 : if $x \sqsubseteq y$ and $x \in U$, open, then $y \in U$.

4.3. LEMMA. A mapping $f: D \rightarrow D'$ is continuous $\Leftrightarrow f(\bigsqcup X) = \bigsqcup f(X)$

for all directed $X \subset D$ (where $f(X) = \{f(x) \mid x \in X\}$ and the second \sqcup is to be taken in D').

PROOF. (\Rightarrow) Let f be continuous. Suppose $x \sqsubseteq y$ in order to show $f(x) \sqsubseteq f(y)$. If not, then $f(x) \in U_{f(y)}$, so $x \in f^{-1}(U_{f(y)})$ which is open. Therefore $y \in f^{-1}(U_{f(y)})$, i.e. $f(y) \in U_{f(y)}$, a contradiction. Hence f is monotonic. It follows that, since $\sqcup X \supseteq X$, $f(\sqcup X) \supseteq f(X)$. Therefore $f(\sqcup X) \supseteq \sqcup f(X)$. If $f(\sqcup X) \not\supseteq \sqcup f(X)$, then $f(\sqcup X) \in U_{\sqcup f(X)}$ and a contradiction can be obtained as above, using condition (ii) for open sets.

(\Leftarrow) Again f is monotonic, since if $x \sqsubseteq y$, then $y = x \sqcup y$, hence $f(y) = f(x) \sqcup f(y)$, so $f(x) \sqsubseteq f(y)$. Therefore if $U \subset D'$ is open, so is $f^{-1}(U) \subset D$. \square

4.4. DEFINITION. If $f(\sqcup X) = \sqcup f(X)$ holds for arbitrary X , then f is called *distributive*.

4.5. COROLLARY. Let $f_i : D \rightarrow D'$ be a collection of continuous mappings. Define $f = \lambda x. \sqcup_i f_i(x)$. Then f is continuous.

PROOF. $f(\sqcup X) = \sqcup_i \sqcup_{x \in X} f_i(x) = \sqcup_{x \in X} \sqcup_i f_i(x) = \sqcup f(X)$. \square

4.6. DEFINITION. $D \times D'$ is the cartesian product partially ordered by $\langle x, x' \rangle \sqsubseteq \langle y, y' \rangle$ iff $x \sqsubseteq x'$ and $y \sqsubseteq y'$. $[D \rightarrow D']$ is the set of continuous maps $\in D \rightarrow D'$ partially ordered by $f \sqsubseteq g \Leftrightarrow \forall x \in D \ f(x) \sqsubseteq g(x)$. Then $D \times D'$ and $[D \rightarrow D']$ are complete lattices with $\sqcup X = \langle \sqcup (X)_0, \sqcup (X)_1 \rangle$ for $X \subset D \times D'$ and $\sqcup F = \lambda x. \sqcup \{f(x) \mid f \in F\}$ for $F \subset [D \rightarrow D']$. (If $z = \langle x, y \rangle$, $z_0 = x$, $z_1 = y$; $\sqcup F$ is continuous by 4.5.) The induced topology on $D \times D'$ is not necessarily the product of the induced topologies on D and D' .

4.7. LEMMA. $f : D \times D' \rightarrow D''$ is continuous $\Leftrightarrow f$ is continuous in each of its variables separately (i.e. $\lambda d. f(d, d'_0)$ and $\lambda d'. f(d_0, d')$ are continuous for all d_0, d'_0).

PROOF. (\Rightarrow) As ever.

(\Leftarrow) $f(\sqcup X) = f(\sqcup X_0, \sqcup X_1)$

$$= \sqcup_{d \in X_0} f(d, \sqcup X_1) = \sqcup_{d \in X_0} \sqcup_{d' \in X_1} f(d, d') = \sqcup_{\langle d, d' \rangle \in X} f(d, d') = \sqcup f(X). \quad \square$$

4.8. LEMMA (Continuity of application). Define Ap (application) $[D \rightarrow D'] \times D \rightarrow D'$ by $\text{Ap}(f, x) = f(x)$. Then Ap is continuous.

PROOF. $\lambda x. f_0(x) = f_0$ is continuous. $\lambda f. f(x_0) = h_0$ with $h_0(\bigsqcup F) = \bigsqcup F(x_0) = \bigsqcup_{f \in F} f(x_0) = \bigsqcup h_0(F)$. Hence h_0 is continuous. Therefore, by 4.7, Ap is continuous. \square

4.9. LEMMA (Continuity of abstraction). Let $f \in [D \times D' \rightarrow D'']$. Define $g_f(x) = \lambda y \in D'. f(x, y)$. Then

- (i) g_f is continuous,
- (ii) $\lambda f. g_f : [D \times D' \rightarrow D''] \rightarrow [D \rightarrow [D' \rightarrow D'']]$ is continuous.

PROOF. (i) $g_f(\bigsqcup X) = \lambda y. f(\bigsqcup X, y) = \lambda y. \bigsqcup_{x \in X} f(x, y) = \bigsqcup_{x \in X} \lambda y. f(x, y) = \bigsqcup g_f(X)$. ($\lambda y. \bigsqcup = \bigsqcup \lambda y$ follows from the definition of \bigsqcup in a function space.)

(ii) Let $L = \lambda f. g_f$. $L(\bigsqcup F) = \lambda x. \lambda y. \bigsqcup F(x, y) = \lambda x. \lambda y. \bigsqcup_{f \in F} f(x, y) = \bigsqcup_{f \in F} \lambda x. \lambda y. f(x, y) = \bigsqcup L(F)$. \square

4.10. DEFINITION. Let $\phi : D \rightarrow D'$, $\psi : D' \rightarrow D$. $\langle \phi, \psi \rangle$ is a *projection* of D on D' iff

- (i) ϕ, ψ are continuous,
- (ii) $\forall x \in D \psi(\phi(x)) = x$,
- (iii) $\forall x \in D' \phi(\psi(x)) \sqsubseteq x$.

4.11. DEFINITION (Construction of D_∞). Let D be an arbitrary nontrivial complete lattice. Define $D_0 = D$, $D_{n+1} = [D_n \rightarrow D_n]$. Mappings $\phi_n : D_n \rightarrow D_{n+1}$ and $\psi_n : D_{n+1} \rightarrow D_n$ are defined as follows:

$$\phi_0(x) = \lambda y \in D_0. x, \quad \psi_0(x') = x'(\perp), \quad \text{where } \perp \in D_0.$$

$$\phi_{n+1}(x) = \phi_n \circ x \circ \psi_n, \quad \psi_{n+1}(x') = \psi_n \circ x' \circ \phi_n \quad (\text{see Diagram 1}).$$

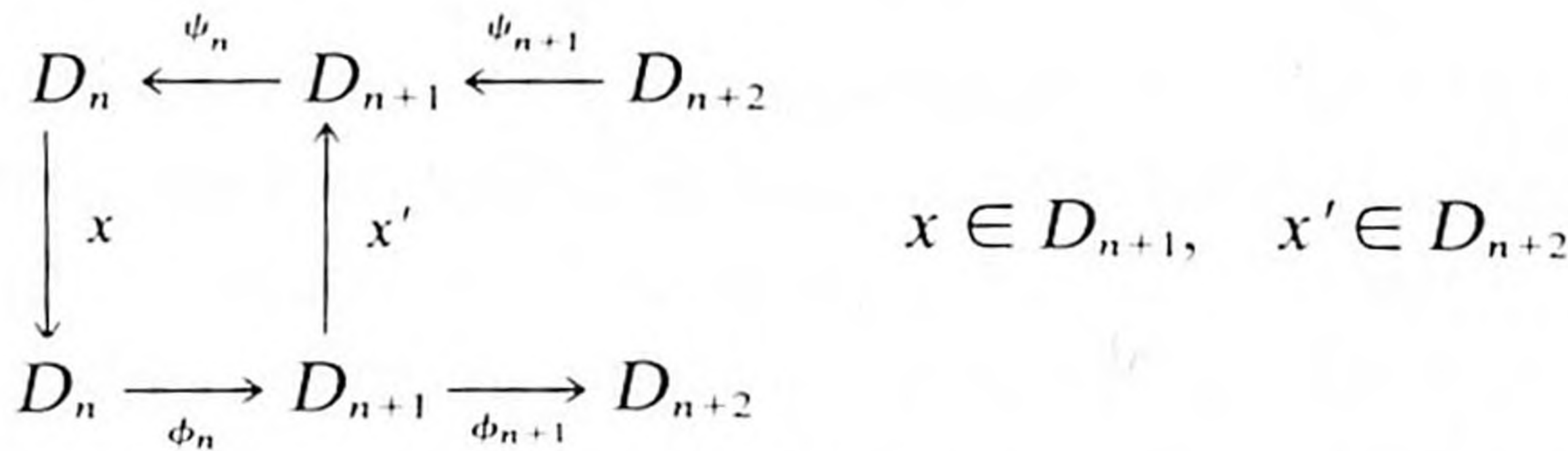


Diagram 1.

By a straightforward induction on n it follows that $\langle \phi_n, \psi_n \rangle$ is a projection of D_{n+1} on D_n . Moreover ϕ_n and ψ_n are distributive. For an element $x = \langle x_n \rangle_{n=0}^\infty$ of the product $\prod_{n=0}^\infty D_n$, x_i denotes the i -th coordinate,

$$D_\infty = \varprojlim (D_n, \psi_n) = \left\{ x \in \prod_n D_n \mid \forall n \in \omega \psi_n(x_{n+1}) = x_n \right\}.$$

For $x, y \in D_\infty$, a partial ordering is defined by $x \sqsubseteq y \Leftrightarrow \forall n \in \omega x_n \sqsubseteq y_n$. Then D_∞ is a complete lattice with for $X \subset D_\infty$, $\sqcup X = \langle \sqcup X_n \rangle_{n=0}^\infty$. This belongs to D_∞ since $\psi_n(\sqcup X_{n+1}) = \sqcup \psi_n(X_{n+1}) = \sqcup X_n$, by the distributivity of ψ_n .

4.12. DEFINITION. Mappings $\Phi_{n,m} : D_n \rightarrow D_m$ are defined (by following the arrows in $D_0 \rightrightarrows D_1 \rightrightarrows \dots$). If $n \leq m$, say $m = n + k$, Φ_{nm} is defined by induction on k . $\Phi_{nn} = \lambda x \in D_n. x$. $\Phi_{n(m+1)} = \phi_m \circ \Phi_{nm}$. If $m \leq n$, say $n = m + k$, Φ_{nm} is again defined by induction k : $\Phi_{(n+1)m} = \Phi_{nm} \circ \psi_n$.

$\Phi_{n\infty} : D_n \rightarrow D_\infty$ is defined by $\Phi_{n\infty}(x) = \langle \Phi_{ni}(x) \rangle_{i=0}^\infty$. $\Phi_{\infty n} : D_\infty \rightarrow D_n$ is defined by $\Phi_{\infty n}(x) = x_n$.

4.13. LEMMA. (i) For $0 \leq n \leq m \leq \infty$, $\langle \Phi_{nm}, \Phi_{mn} \rangle$ is a projection of D_n on D_m .
(ii) For $0 \leq n \leq m \leq l \leq \infty$, $\Phi_{ml} \circ \Phi_{nm} = \Phi_{nl}$.

PROOF. Standard. \square

It follows that up to isomorphism $D_0 \subseteq D_1 \subseteq \dots \subseteq D_\infty$. In fact in the category of *complete* lattices with continuous mappings as morphisms, D_∞ is not only the inverse limit $\varprojlim (D_n, \psi_n)$, but also a direct limit: $D_\infty = \varinjlim (D_n, \phi_n)$. Note however $D_\infty \neq \bigcup_n D_n$, D_∞ is the completion of $\bigcup D_n$. Henceforth each element $x \in D_n$ will be identified with $\Phi_{n\infty}(x) \in D_\infty$, as is customary with direct limits. In particular

4.14. LEMMA. (i) If $x \in D_n$, then $x = x_n$.
(ii) If $x \in D_n$, then $\phi_n(x) = x$.
(iii) If $x \in D_{n+1}$, then $\psi_n(x) \sqsubseteq x$.

PROOF. (i) x in D_∞ is $\langle \dots, \psi(x), x, \phi(x), \phi(\phi(x)), \dots \rangle$. Hence x_n is x .

(ii) $\phi_n(x)$ in D_∞ is $\langle \dots, \psi(\phi_n(x)), \phi_n(x), \phi(\phi_n(x)), \dots \rangle$. Since $\psi(\phi(x)) = x$, this is the same as x in D_∞ .

(iii) Analogous, using $\phi(\psi(x)) \sqsubseteq x$. \square

Due to the identifications, properties of D_∞ are more elegant to formulate:

4.15. LEMMA. In D_∞

- (i) $(x_n)_m = x_{\min(n, m)}$.
- (ii) If $n \leq m$, then $x_n \sqsubseteq x_m \sqsubseteq x$.
- (iii) $x = \bigsqcup_{n=0}^{\infty} x_n$.
- (iv) \top_n and \perp_n are top and bottom of D_n .
- (v) $\perp_n = \perp$; $\top_n = \top$.

PROOF. (i) If $m \leq n$, $(x_n)_m = \Phi_{nm}x_n = \psi \circ \dots \circ \psi(x_n) = x_m$, since $x \in D_{\infty}$. If $m \geq n$, $(x_n)_m = \phi \circ \dots \circ \phi(x_n) = x_n$ by 4.14(ii).

(ii) First, by 4.14(iii), $x_m \sqsubseteq x_{m+1}$, since $x_m = \psi_m(x_{m+1})$. Hence $x_0 \sqsubseteq x_1 \sqsubseteq \dots$. Furthermore $x_n \sqsubseteq x$ since $\forall i (x_n)_i = x_{\min(n, i)} \sqsubseteq x_i$.

(iii) $\bigsqcup_n x_n = \langle \bigsqcup_n (x_n)_i \rangle_{i=0}^{\infty} = \langle \bigsqcup_n x_{\min(n, i)} \rangle_{i=0}^{\infty} = \langle x_i \rangle_{i=0}^{\infty} = x$.

(iv) Let \top'_n and \perp'_n be resp. top and bottom of D_n . Then

$$\top = \bigsqcup D_{\infty} = \langle \bigsqcup D_n \rangle_{n=0}^{\infty} = \langle \top'_n \rangle_{n=0}^{\infty}$$

and

$$\perp = \bigsqcup \emptyset = \langle \bigsqcup \emptyset \rangle_{n=0}^{\infty} = \langle \perp'_n \rangle_{n=0}^{\infty}.$$

Hence $\top_n = \top'_n$, $\perp_n = \perp'_n$.

(v) By (ii), $\perp_n \sqsubseteq \perp$ and $\perp \sqsubseteq \perp_n$ always. So $\perp_n = \perp$. On the other hand it follows by induction that $\phi_n(\top_n) = \top_{n+1}$, since $\top_{n+1} = \lambda x \in D_n. \top_n$. Moreover since $\langle \top_n \rangle_{n \in \omega} \in D_{\infty}$, $\psi_n(\top_{n+1}) = \top_n$. Therefore \top_n in D_{∞} , i.e., $\langle \Phi_{nm}(\top_n) \rangle_{m \in \omega}$ is $\langle \top_m \rangle_{m \in \omega} = \top$. \square

4.16. DEFINITION. In D_{∞} one can define a binary operation application: $x \cdot y = \bigsqcup_n x_{n+1}(y_n)$. On the RHS the application is the usual one $D_{n+1} \times D_n \rightarrow D_n$ and the \bigsqcup is to be taken in D_{∞} after identification.

4.17. LEMMA. Application is well defined, in the sense that if $x_{n+1} \in D_{n+1}$ and $y_n \in D_n$, then $x_{n+1} \cdot y_n = x_{n+1}(y_n)$.

PROOF.

$$\begin{aligned} x_{n+1} \cdot y_n &= \bigsqcup_{i=0}^{\infty} (x_{n+1})_{i+1}((y_n)_i) \\ &= \bigsqcup_{i=0}^n x_{i+1}(y_i) \quad \text{by 4.15(i),} \\ &= x_{n+1}(y_n) \quad \text{by 4.15(ii).} \quad \square \end{aligned}$$

4.18. LEMMA. Application is continuous.

PROOF. $x \cdot y = \bigsqcup_n \Phi_{n\infty}(x_{n+1}(y_n))$. Now apply 4.5, 4.8 and 4.13. \square

4.19. LEMMA. (i) $x_{n+1} \cdot y = x_{n+1} \cdot y_n = (x \cdot y_n)_n$.
(ii) $x_0 \cdot y = x_0 = (x \cdot \perp)_0$.

PROOF. (i) First it is shown by induction on $i \geq n$ that $(x_{n+1})_{i+1}(y_i) = x_{n+1}(y_n)$. For $i = n$ this is clear. Now

$$\begin{aligned} (x_{n+1})_{i+2}(y_{i+1}) &= \phi_{i+1}((x_{n+1})_{i+1})(y_{i+1}) = \phi_i \circ (x_{n+1})_{i+1} \circ \psi_i(y_{i+1}) \\ &= \phi_i((x_{n+1})_{i+1}(y_i)) = (x_{n+1})_{i+1}(y_i), \quad \text{by 4.14(ii),} \\ &= x_{n+1}(y_n) \end{aligned}$$

by the induction hypothesis. Therefore

$$\begin{aligned} x_{n+1} \cdot y &= \bigsqcup_{i=n}^{\infty} (x_{n+1})_{i+1}(y_i) \\ &= \bigsqcup_{i=n}^{\infty} x_{n+1}(y_n) = x_{n+1} \cdot y_n \quad (\text{by 4.17}). \end{aligned}$$

Again by induction on $i \geq n$ it is shown that $(x_{i+1}(y_n))_i = x_{n+1}(y_n)$. For $i = n$ this is clear. Now

$$\begin{aligned} (x_{i+2}(y_n))_{i+1} &= \Phi_{i+1n}(x_{i+2}(\phi_i((y_n)_i))) = \Phi_{in} \circ \psi_i \circ x_{i+2}(\phi_i(y_n)_i) \\ &= \Phi_{in}((\psi_{i+1}(x_{i+2}))(y_n)_i) = \Phi_{in}(x_{i+1}(y_n)_i) \\ &= (x_{i+1}(y_n))_i = x_{n+1}(y_n) \end{aligned}$$

by the induction hypothesis. Therefore

$$(x \cdot y_n)_n = \left(\bigsqcup_i x_{i+1}((y_n)_i) \right)_n = \bigsqcup_i (x_{i+1}((y_n)_i))_n = \bigsqcup_i x_{n+1}(y_n) = x_{n+1} \cdot y_n.$$

(ii) By (i), $x_0 \cdot y = (x_0)_1 \cdot y = (x_0)_1(y_0) = \phi_0(x_0)(y_0) = x_0$. Also $x_0 = \psi_0(x_1) = x_1(\perp_0) = (x \cdot \perp)_0 = (x \cdot \perp)_0$. \square

4.20. THEOREM (extensionality). For $x, y \in D_\infty$,

- (i) $x \sqsubseteq y \Leftrightarrow \forall z \in D_\infty x \cdot z \sqsubseteq y \cdot z$,
- (ii) $x = y \Leftrightarrow \forall z \in D_\infty x \cdot z = y \cdot z$.

PROOF. (i) (\Rightarrow) By monotonicity of $\lambda x(x \cdot z)$. (\Leftarrow) Suppose $\forall z x \cdot z \sqsubseteq y \cdot z$. Then $x \cdot \perp \sqsubseteq y \cdot \perp$, so $x_0 = (x \cdot \perp)_0 \sqsubseteq (y \cdot \perp)_0 = y_0$ by 4.19(ii). Moreover $x \cdot z_n \sqsubseteq y \cdot z_n$, so $x_{n+1}(z_n) = (x \cdot z_n)_n \sqsubseteq (y \cdot z_n)_n = y_{n+1}(z_n)$ by 4.17 and 4.19(i). Hence $x_{n+1} \sqsubseteq y_{n+1}$. Now we have $\forall n x_n \sqsubseteq y_n$, i.e. $x \sqsubseteq y$.

(ii) Immediate by (i). \square

4.21. THEOREM (completeness). $\forall f \in [D_\infty \rightarrow D_\infty] \exists x \in D_\infty f(y) = x \cdot y$.

PROOF. Let $x = \bigsqcup_n (\lambda y \in D_n. (f(y))_n)$. Note that this is the supremum of a directed set. Remark that for a_{ij} in a complete lattice,

$$\bigsqcup_{i,j} a_{ij} = \bigsqcup_k a_{kk} \quad \text{if } \forall i, j \exists k a_{ij} \sqsubseteq a_{kk}.$$

Now

$$\begin{aligned} x \cdot y &= \bigsqcup_m x_{m+1}(y_m) = \bigsqcup_m (x \cdot y_m)_m = \bigsqcup_m \left(\left(\bigsqcup_n \lambda y \in D_n. (f(y))_n \right) \cdot y_m \right)_m \\ &= \bigsqcup_{m,n} (\lambda y \in D_n (f(y))_n \cdot y_m)_m = \bigsqcup_m (\lambda y \in D_m (f(y))_m \cdot y_m) \\ &= \bigsqcup_m (f(y_m))_m = \bigsqcup_{k,l} (f(y_k))_l = \bigsqcup_k f(y_k) = f(y). \end{aligned}$$

Comments that should accompany these equations follow easily from the continuity (monotonicity) of the functions involved and the remark above. \square

4.22. COROLLARY. D_∞ is homeomorphic to $[D_\infty \rightarrow D_\infty]$.

PROOF. For $x \in D_\infty$ let $F(x) = \lambda y \in D_\infty x \cdot y$. F is surjective by 4.21, injective by 4.20, continuous by 4.9(i). The inverse to F is, by the proof of 4.21, $\lambda f \bigsqcup_n (\lambda y \in D_n (f(y))_n)$ which is continuous by 4.9(ii) and 4.5. \square

4.23. THEOREM. D_∞ is an extensional λ -algebra.

PROOF. Combinatory completeness follows from 4.21 since application and abstraction are continuous. Extensionality was proved in 4.20(ii). Hence 1.16(2) applies. \square

Since D_∞ is extensional there is no ambiguity interpreting λ -terms in it. However for later reference, the interpretation will be given explicitly.

4.24. DEFINITION. (i) A valuation (in D_∞) is a mapping $\rho: \text{variables} \rightarrow D_\infty$.

(ii) For $d \in D_\infty$ and x a variable, $\rho(d/x)$ is the valuation ρ' with $\rho'(y) = \rho(y)$ if $y \neq x$ then d .

(iii) The interpretation of M in D_∞ under ρ , $\llbracket M \rrbracket \rho$, is defined inductively: $\llbracket x \rrbracket \rho = \rho(x)$, $\llbracket MN \rrbracket \rho = \llbracket M \rrbracket \rho \llbracket N \rrbracket \rho$, $\llbracket \lambda x. M \rrbracket \rho = \lambda d. \llbracket M \rrbracket \rho(d/x) \in D_\infty$ after identification with $[D_\infty \rightarrow D_\infty]$.

Roughly, in the interpretation formal variables are replaced by variables

ranging over D_∞ and application and abstraction are to be taken in D_∞ . From this it should be obvious that the interpretation is correct. This can be made precise by showing inductively $\llbracket (\lambda x M) N \rrbracket \rho = \llbracket M \rrbracket \rho (\llbracket N \rrbracket \rho / x) = \llbracket M(x/N) \rrbracket \rho$.

Whenever possible the valuation ρ will be omitted in the notation $\llbracket M \rrbracket$.

5. Solvability

The concept of solvability and the related notion of head normal form were introduced respectively in the dissertations of Barendregt and Wadsworth. Both theses give arguments for the computational irrelevance of unsolvable terms. Therefore a λ -theory (or λ -algebra) is called *sensible* iff it equates all unsolvable terms. It turned out that there is a unique maximal sensible theory. In Section 7 it will be proved that this theory equals $\text{Th}(D_\infty)$.

5.1. DEFINITION. (i) A closed term M is *solvable* iff $\lambda \vdash MN_1 \cdots N_n = I$ for some n and terms $N_1 \cdots N_n$.

(ii) An arbitrary term M is solvable iff its closure $\lambda \tilde{x}. M$ is solvable. M is *unsolvable* iff M is not solvable.

(iii) $\mathcal{H} = \{M = N \mid M, N \text{ unsolvable}\}$.

To see the particular role of I in this definition, note that M (closed) is solvable iff $\forall P \exists \tilde{N} M\tilde{N} = P$.

5.2. DEFINITION. (i) Each λ -term M is of the form

$$\lambda x_1 \cdots x_n. (\lambda x. P) Q M_1 \cdots M_m \quad \text{or} \quad \lambda x_1 \cdots x_n. x_i M_1 \cdots M_m,$$

$m, n \geq 0$. In the first case $(\lambda x. P)Q$ is the *head redex* of M . In the second case x_i is the *head variable* of M and M is said to be in *head normal form* (hnf).

The following can be proved syntactically. A semantic proof will be given in 7.9 and 7.10.

5.3. THEOREM. (i) M is solvable $\Leftrightarrow M$ has a hnf.

(ii) $\lambda\eta + \mathcal{H}$ is consistent.

5.4. EXAMPLES. I and Y are solvable; Ω and $K^\infty \equiv YK$ are unsolvable

(note that $\lambda \vdash Y(KI) = I$, $\Omega N_1 \cdots N_n \rightarrow M \Rightarrow M \equiv \Omega N'_1 \cdots N'_n$ with $N_i \rightarrow N'_i$, and $\lambda \vdash K^\infty M = K^\infty$ for all M). Note also $\lambda + \mathcal{H} \vdash \lambda x. \Omega = \Omega x = \Omega$, since M unsolvable $\Rightarrow MN, \lambda x. M$ unsolvable.

5.5. DEFINITION. (i) A context $C[\cdot]$ is a term with some holes in it. More formally: any variable x is a context; $[\cdot]$ is a context; if $C_1[\cdot], C_2[\cdot]$ are contexts, so are $(C_1[\cdot] C_2[\cdot])$ and $(\lambda x. C[\cdot])$. If M is an arbitrary term and $C[\cdot]$ a context, $C[M]$ denotes the result of placing M in the holes of $C[\cdot]$. In this act, free variables of M may become bound in $C[M]$.

(ii) M and N are *solvably equivalent*, notation $M \sim_s N$ iff $\forall C[\cdot] [C[M]$ is solvable $\Leftrightarrow C[N]$ is solvable].

(iii) $\mathcal{H}^* = \{M = N \mid M, N \in \Lambda^0 \text{ and } M \sim_s N\}$.

5.6. LEMMA. $\mathcal{H}^* = \mathcal{H}^{*+}$.

PROOF. Induction on the length of proof shows that $\lambda + \mathcal{H}^* \vdash M = N \Rightarrow M \sim_s N$. \square

5.7. COROLLARY. \mathcal{H}^* is consistent hence a λ -theory.

PROOF. $I \not\sim_s \Omega$, hence $I = \Omega \notin \mathcal{H}^*$. \square

5.8. LEMMA. If $\mathcal{H} + M = N$ is consistent, then $\mathcal{H}^* \vdash M = N$.

PROOF. First note that $\{I = K^\infty\}$ is inconsistent: $\lambda + I = K^\infty \vdash M = IM = K^\infty M = K^\infty$ for all M . Now suppose $M = N \notin \mathcal{H}^*$. Then, say, $C[M]$ is solvable and $C[N]$ is unsolvable for some $C[\cdot]$. Therefore $\lambda \vdash (\lambda \vec{x}. C[M])\vec{P} = I$ for some \vec{P} and $\mathcal{H} \vdash C[N] = K^\infty$. Now

$$\begin{aligned} \lambda + \mathcal{H} + M = N \vdash I &= (\lambda \vec{x}. C[M])\vec{P} = (\lambda \vec{x}. C[N])\vec{P} \\ &= (\lambda \vec{x}. K^\infty)\vec{P} = K^\infty \vec{P} = K^\infty, \end{aligned}$$

so $\lambda + \mathcal{H} + M = N$ would be inconsistent. \square

5.9. COROLLARY. (i) \mathcal{H}^* is the unique maximal λ -theory extending \mathcal{H} .

(ii) Moreover, \mathcal{H}^* proves extensionality.

PROOF. (i) Since $\text{Con}(\mathcal{H})$, by 5.3, $\mathcal{H} \subset \mathcal{H}^*$ by 5.8. If T is a consistent extension of \mathcal{H} , then for each $M = N \in T$, $M = N \in \mathcal{H}^*$ by 5.8; so $T \subset \mathcal{H}^*$.

Therefore \mathcal{H}^* contains each consistent extension of \mathcal{H} and being itself consistent, 5.7, the statement follows.

(ii) $\lambda + \mathcal{H} + (\lambda x. Mx) = M$ ($x \notin \text{FV}(M)$) is consistent by 5.3(ii). Hence $\mathcal{H}^* \vdash \lambda x. Mx = M$. \square

The possibility that \mathcal{H} has a *unique* maximally consistent extension is due to the fact that the language of the λ -calculus is logic free, i.e. it is not possible that for an undecided sentence σ we make two extensions by adding σ and $\neg\sigma$ respectively, because the language does not contain negation. The theory λ however has 2^{\aleph_0} maximal consistent extensions.

5.10. DEFINITION. A λ -algebra \mathfrak{M} is *sensible* iff $\mathfrak{M} \models \mathcal{H}$. In that case $\text{Th}(\mathfrak{M}) \subset \mathcal{H}^*$ by 5.9.

The “least” sensible model is $\mathfrak{M}^0(\mathcal{H}^*)$:

5.11. COROLLARY. $\mathfrak{M}^0(\mathcal{H}^*)$ is algebraically simple, i.e. has no proper homomorphic images.

PROOF. If \mathfrak{M} were a proper image of $\mathfrak{M}^0(\mathcal{H}^*)$, then $\text{Th}(\mathfrak{M})$ would be a proper extension of \mathcal{H}^* . \square

In Section 7 it will be shown that $\mathfrak{M}^0(\mathcal{H}^*)$ is the interior of D_∞ .

6. Böhm trees

The trees introduced in this section are inspired by the proof of the Theorem of Böhm, 2.23, and the concept of solvability. The Böhm trees are useful for the analysis of $\mathcal{P}\omega$ and D_∞ . See NAKAJIMA [1975] for a related family of trees.

6.1. DEFINITION. A *tree* is a set A of sequence numbers such that

- (i) if $\alpha \in A$ and $\beta < \alpha$ (ordering of sequence numbers), then $\beta \in A$,
- (ii) for each $\alpha \in A$ there are only finitely many immediate successors of α in A . The $\alpha \in A$ are called the *nodes* of the tree. The depth of $\alpha = \langle n_0, \dots, n_{k-1} \rangle$, notation $d(\alpha)$, is k . $*$ denotes concatenation, i.e. $\langle \vec{n} \rangle * \langle \vec{m} \rangle = \langle \vec{n}, \vec{m} \rangle$. Then *subtree* at α of A , A_α , is the set $\{\beta \mid \alpha * \beta \in A\}$.

6.2. DEFINITION. (i) A *labelled tree* is a tree where at each node there is a

symbol which is either Ω or $\lambda x_1 \cdots x_n. x_i$ for some variables x_1, \dots, x_n, x_i . To be precise, a labelled tree is a mapping f_A from the sequence numbers into the set

$$\{*, \Omega\} \cup \{\langle i_1, \dots, i_n \rangle, i \mid i, n, i_1, \dots, i_n \in \omega\},$$

such that $A = \{\alpha \mid f_A(\alpha) = *\}$ is a tree. $f_A(\alpha) = \langle i_1, \dots, i_n \rangle, i$ (resp. Ω) means that $\lambda a_{i_1} \cdots a_{i_n}. a_i$ (resp. Ω) is written at node $\alpha \in A$.

(ii) If A, B are labelled trees, then

$$A \equiv_k B \Leftrightarrow \forall \alpha \left[[d(\alpha) < k \Rightarrow f_A(\alpha) = f_B(\alpha)] \wedge [d(\alpha) = k \Rightarrow f_A(\alpha) \neq * \Leftrightarrow f_B(\alpha) \neq *] \right].$$

i.e. for depth $< k$ the labelled trees are equal and the nodes of depth $k - 1$ have in both trees the same number of successors.

6.3. DEFINITION. The *Böhm tree* of a λ -term M , $BT(M)$, is a labelled tree defined as follows: If m is unsolvable, then $BT(M) = \Omega$. If M is solvable, say M has hnf $\lambda x_1 \cdots x_n. x_i M_1 \cdots M_m$, then $BT(M)$ is

$$\lambda x_1 \cdots x_n. x_i \overbrace{BT(M_1) \cdots BT(M_m)}.$$

To be precise, if M is unsolvable, then $BT(M)(\langle \cdot \rangle) = \Omega$, $BT(M)(\langle j \rangle * \alpha) = *$. If M is solvable, say M has hnf $\lambda a_{i_1} \cdots a_{i_n}. a_i M_0 \cdots M_{m-1}$, then $BT(M)(\langle \cdot \rangle) = \langle i_1, \dots, i_n \rangle, i$, $BT(M)(\langle j \rangle * \alpha) = BT(M_j)(\alpha)$, for $j < m$ and $BT(M)(\langle j \rangle * \alpha) = *$ for $j \geq m$.

Free and bound occurrences of a variable in a Böhm tree are defined as for terms. As with terms, Böhm trees are considered modulo a change of bound variables.

From the Church–Rosser theorem it follows that if M has a hnf $\lambda x_1 \cdots x_n. x_i M_1 \cdots M_m$, then n, i, m are uniquely determined. Hence the Böhm tree of M is well defined and if $\lambda \vdash M = N$, then $BT(M) = BT(N)$.

6.4. EXAMPLE.

$$\begin{array}{llll} BT(S): & \lambda abc. a. & BT(Sx\Omega): & \lambda x. x. \\ & \begin{array}{c} \diagup \quad \diagdown \\ c \quad b \\ \quad \quad | \\ \quad \quad c \end{array} & & \begin{array}{c} \diagup \quad \diagdown \\ c \quad \Omega \end{array} \\ & & & BT(Y): & \lambda f. f. \\ & & & & \begin{array}{c} | \\ f \\ | \\ \vdots \end{array} \end{array}$$

Note that although $\lambda x.x$ and $\lambda xy.xy$ are extensionally equal, they have different Böhm trees. Therefore the following equivalence relation is introduced.

6.5. DEFINITION. (i) M', N' merge $BT(M), BT(N)$ up to k iff $\lambda\eta \vdash M = M', \lambda\eta \vdash N = N'$ and $BT(M') \equiv_k BT(N')$.

(ii) M, N have equivalent Böhm trees, notation $BT(M) \sim_\eta BT(N)$, iff $\forall k \exists M', N' M', N'$ merge $BT(M), BT(N)$ up to k .

Now we give some examples of terms with equal or equivalent Böhm trees.

6.6. Example. (i) By the fixed point theorem there exists a term A such that $Ax \rightarrow \lambda z.z(Ax)$. Then $BT(Ax) = BT(Ay)$ (x and y disappear from the tree).

(ii) Let $Y_z = \lambda f.(\lambda xz.f(xxz))(\lambda xz.f(xxz))z$. Then Y_z is an alternative fixed point operator not convertible with Y . But $BT(Y_z) = BT(Y)$.

(iii) $BT(\lambda x.x) \sim_\eta BT(\lambda xy.xy)$.

The following is a less trivial example of equivalent Böhm trees. Together with the characterization Theorem 7.1 it shows that in D_∞ a normal form may be equal to a term without a nf.

6.7. EXAMPLE (Wadsworth). Let $J = Y(\lambda jxy.x(jy))$. Then $BT(J) \sim_\eta BT(I)$.

PROOF. The hnf of J is $\lambda x_0x_1.x_0(Jx_1)$, so $BT(J)$ is

$$\begin{array}{c} \lambda x_0x_1.x_0 \\ | \\ \lambda x_2.x_1 \\ | \\ \lambda x_3.x_2 \\ | \\ \vdots \end{array}$$

This can be merged to any depth with $BT(I)$ by some η expansions (i.e. the opposite of a contraction) of the latter. \square

6.8. THEOREM. $\mathcal{H}^* \vdash M = N \Rightarrow BT(M) \sim_\eta BT(N)$.

The rather technical proof occupies the rest of this section and can be omitted at a first reading.

6.9. DEFINITION. Let A, A' be Böhm trees.

(i) A, A' are *top mergeable* iff either A, A' are both Ω or A and A' are

$$\begin{array}{ccc} \lambda x_1 \cdots x_n \cdot x_i & & \lambda x_1 \cdots x_{n'} \cdot x_{i'} \\ & \swarrow \quad \searrow & \\ \square_1 \cdots \square_m & & \square'_1 \cdots \square'_{m'} \end{array}$$

respectively, and $i = i'$ and $n - m = n' - m'$ (possibly negative numbers); the sequences x_1, \dots, x_n and $x_1, \dots, x_{n'}$ can be assumed to start similarly, by a change of bound variables.

(ii) Let α be a common node of A, A' . A, A' are *mergeable at α* iff A_α, A'_α are top mergeable.

(iii) A, A' *separate at depth k* iff $A \equiv_k A'$ and A, A' are not mergeable at some common node α with $d(\alpha) = k$.

6.10. LEMMA. Let $BT(M) \equiv_k BT(N)$. If $BT(M), BT(N)$ are mergeable at all α with $d(\alpha) = k$, then they can be merged up to $k + 1$.

PROOF. Let $d(\alpha) = k$. $BT(M)_\alpha, BT(N)_\alpha$ are, say,

$$\begin{array}{ccc} \lambda x_1 \cdots x_n \cdot x_i & \text{and} & \lambda x_1 \cdots x_{n'} \cdot x_{i'} \\ & \swarrow \quad \searrow & \\ \square_1 \cdots \square_m & & \square'_1 \cdots \square'_{m'} \end{array}$$

Now make an η -change as follows

$$\begin{array}{ccc} \lambda x_1 \cdots x_n x_{n+1} \cdots x_{n+n'} \cdot x_i & & \lambda x_1 \cdots x_n x_{n'+1} \cdots x_{n'+n} \cdot x_i \\ & \swarrow \quad \downarrow \quad \searrow & \\ \square_1 \cdots \square_m x_{n+1} \cdots x_{n+n'} & & \square'_1 \cdots \square'_{m'} x_{n'+1} \cdots x_{n'+n} \end{array}$$

Note that $m + n' = m' + n$ by the mergeability at α , so after the change, α has the same number of successors in both trees. After this change is made for all α with $d(\alpha) = k$, the resulting labelled trees are Böhm trees of M', N' , say, which merge $BT(M), BT(N)$ up to $k + 1$. \square

6.11. COROLLARY. If $BT(M) \not\equiv_n BT(N)$, then $\exists k, M', N'$ such that M', N' merge $BT(M), BT(N)$ up to k and $BT(M'), BT(N')$ separate at depth k .

PROOF. Let k be maximal such that $BT(M), BT(N)$ can be merged up to k , by some M', N' . Then $BT(M') \equiv_k BT(N')$. Suppose that $BT(M'), BT(N')$

are mergeable at all α with $d(\alpha) = k$. Then by 6.10, $BT(M')$, $BT(N')$, and hence $BT(M)$, $BT(N)$, can be merged up to $k + 1$, contradicting the maximality of k . Therefore $BT(M')$, $BT(N')$ separate at depth k . \square

6.12. DEFINITION. (i) A *transformation* is a mapping $f : \Lambda \rightarrow \Lambda$.

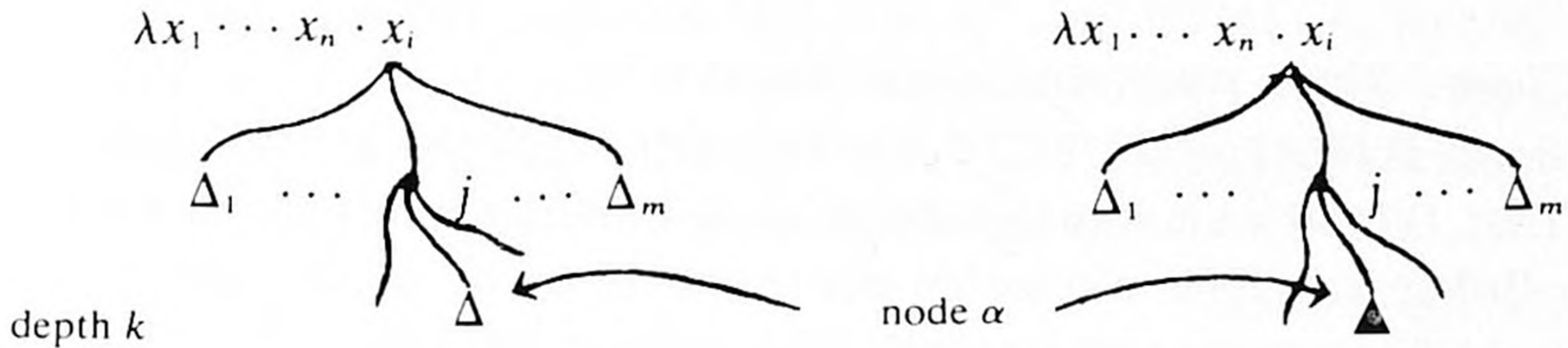
(ii) A *solving transformation* f is either defined by $f(P) = Px$ for some x or by $f(P) = P[x/Nx]$ for some x and closed N .

(iii) A *Böhm transformation* is a finite composition of solving transformations. Notations: π ranges over Böhm transformations; $M^\pi = \pi(M)$.

6.13. DEFINITION. $BT(M)$ is *head original* up to k iff $BT(M)$ has a free head variable which does not occur freely at any other node α with $d(\alpha) \leq k$.

6.14. LEMMA. If $BT(M)$, $BT(N)$ are head original up to k and separate at depth $k > 0$, then for some π , $BT(M^\pi)$, $BT(N^\pi)$ separate at depth $k - 1$.

PROOF. Let the trees separate at node α :



Define $\pi(P) = Px_1 \cdots x_n [x_i / U_j^m x_i]$, where $U_j^m = \lambda y_1 \cdots y_m \cdot y_j$. Then $BT(M^\pi)$, $BT(N^\pi)$ separate at depth $k - 1$:



The assumption of head originality is needed to insure that the difference Δ , \blacktriangle is not lost by the substitution $[x_i / U_j^m x_i]$. \square

6.15. LEMMA. Let $BT(M)$, $BT(N)$ separate at depth $k > 0$. Then for some π , $BT(M^\pi)$ are head original up to k and still separate at depth k .

PROOF. Let $C_q = \lambda z_0 \cdots z_{q+1} \cdot z_{q+1} z_0 \cdots z_q$. For a node α in a Böhm tree let $\# \alpha$ be $\max(s, t)$ where $\lambda a_{i_1} \cdots a_{i_s} \cdot a_j$ is the label at α and t is the number of successors of α ; $\# \alpha$ is 0 if Ω is the label at α . The assumption implies that

M, N have hnf's, say $\lambda x_1 \cdots x_n. x_i M_1 \cdots M_m$ and $\lambda x_1 \cdots x_n. x_i N_1 \cdots N_m$ respectively. Now define

$$\pi(P) = (Px_1 \cdots x_n)[x_i/C_q x_i]z_{m+1} \cdots z_{q+1},$$

where $q > 2(\#\alpha)$ for all α in $BT(M), BT(N)$ with $d(\alpha) \leq k$ and $z_{m+1} \cdots z_{q+1}$ are fresh variables. Note that the Böhm trees at depth $\leq k$ of M^π, N^π result from those of M, N respectively by replacing the tops

$$\begin{array}{c} \lambda x_1 \cdots x_n. x_i \\ \swarrow \quad \searrow \\ \square_1 \cdots \square_m \end{array} \quad \text{by} \quad \begin{array}{c} z_{q+1} \\ \swarrow \quad \downarrow \quad \searrow \\ x_i \square_1 \cdots \square_m z_{m+1} \cdots z_q \end{array}$$

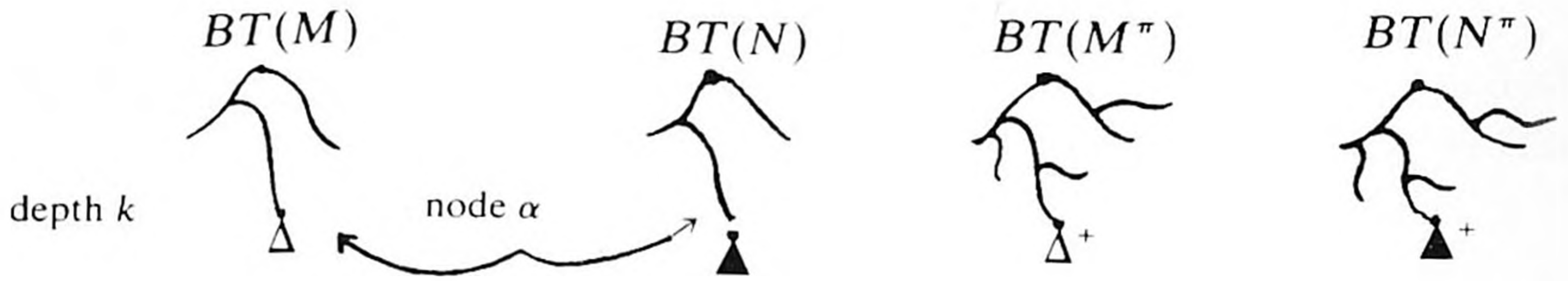
and the internal nodes with free head variable x_i

$$(*) \quad \begin{array}{c} \lambda y_1 \cdots y_s. x_i \\ \swarrow \quad \searrow \\ \square_1 \cdots \square_t \end{array} \quad \text{by} \quad \begin{array}{c} \lambda y_1 \cdots y_s w_{t+1} \cdots w_{q+1}. w_{q+1} \\ \swarrow \quad \downarrow \quad \searrow \\ x_i \square_1 \cdots \square_t w_{t+1} \cdots w_q \end{array}$$

(here $q > t$ is used). Clearly $BT(M^\pi)$ and $BT(N^\pi)$ are head original up to k .

Claim: These trees separate at depth k .

Since $BT(M) \equiv_k BT(N)$, also $BT(M^\pi) \equiv_k BT(N^\pi)$. By assumption $BT(M), BT(N)$ are not mergeable at some node α with $d(\alpha) = k$. Consider the Böhm trees



Suppose Δ and \blacktriangle have both the free head variable x_i . Referring to $(*)$, Δ^+ and \blacktriangle^+ are top mergeable iff $s + (q + 1 - t) - (1 + q) = s' + (q' + 1 - t') - (1 + q')$ iff $s - t = s' - t'$ which is not the case since Δ and \blacktriangle are not top mergeable. Hence $BT(M^\pi), BT(N^\pi)$ are not mergeable at α , i.e. separate at depth k . The same conclusion holds in the other cases (x_i is head variable of Δ , not of \blacktriangle ($q > s$ $s' + t$ insures that Δ^+ and \blacktriangle^+ have different head variables and hence are not top mergeable); x_i is not head variable of either Δ or \blacktriangle). \square

6.16. COROLLARY. *If $BT(M), BT(N)$ separate at depth $k > 0$, then for some π , $BT(M^\pi), BT(N^\pi)$ separate at depth $k - 1$.*

PROOF. By 6.14 and 6.15. \square

6.17. LEMMA. *If $BT(M)$, $BT(N)$ separate at level 0, then for some π , M^π is solvable, N^π is unsolvable, or conversely.*

PROOF. If M is unsolvable, $BT(M) = \Omega$ and hence N is solvable since $BT(M)$, $BT(N)$ differ at level 0. In this case take π the identity. If M, N are solvable, say $M \rightarrow \lambda x_1 \cdots x_n. x_i M_1 \cdots M_m$, $N \rightarrow \lambda x_1 \cdots x_{n'}. x_{i'} N_1 \cdots N_{m'}$, then by assumption $i \neq i'$ or $n - m \neq n' - m'$. In case $i \neq i'$ take $P^\pi = P[x_i / (\lambda y_1 \cdots y_m. I)x_i][x_{i'} / \Omega x_{i'}]$. Then $M^\pi = I$ and $N^\pi = \lambda x_1 \cdots x_n. \Omega \cdots = \Omega \pmod{\mathcal{H}}$ and hence unsolvable. In case $i = i'$ and $n - m \neq n' - m'$, let $P^{\pi_0} = Px_1 \cdots x_p \Omega$ with p large enough ($\geq \max(m, n, m', n')$). Then $M^{\pi_0} = x_i M_1 \cdots M_m x_{n+1} \cdots x_p \Omega$. $N^{\pi_0} = x_{i'} N_1 \cdots N_{m'} x_{n'+1} \cdots x_p \Omega$. The length of the sequences after x_i is $m + p - n + 1$, $m' + p - n' + 1$ respectively which differ since $n - m \neq n' - m'$. Hence by defining $\pi = \pi_1 \circ \pi_0$, with $P^{\pi_1} = P[x_i / U_j^q x_i]$ where $U_j^q = \lambda y_1 \cdots y_n. y_j$ is an appropriate selector, the required π is found. \square

PROOF OF THEOREM 6.8. Suppose $M = N \in \mathcal{H}^*$, but $BT(M) \not\sim_\eta BT(N)$. By 6.11 for some M', N' , $\lambda \eta \vdash M = M'$, $N = N'$ and $BT(M')$, $BT(N')$ separate at depth k . By iterated application of 6.16, $BT(M'^{\pi_0})$, $BT(N'^{\pi_0})$ separate at depth 0 for some π_0 . Hence it follows from 6.17 that for some π_1 , M'^{π_1} is, say, solvable and N'^{π_1} is unsolvable. Now for each π there is a $C_\pi[\cdot]$ such that $\pi(P) = C_\pi[P]$ (a substitution $M[x/Nx]$ can be written as $C[M]$ with $C[\cdot] = (\lambda x. [\cdot])(Nx)$). So $C_{\pi_1}[M']$ is solvable and $C_{\pi_1}[N']$ is unsolvable, hence $\mathcal{H}^* \not\vdash M' = N'$ by the proof of 5.6. But this contradicts $\mathcal{H}^* \vdash M = N$, since \mathcal{H}^* proves extensionality (see 5.9(ii)). \square

7. Analysis of D_∞

Using Böhm trees it is possible to give an elegant characterization of equality of terms in $\mathcal{P}\omega$ and D_∞ .

7.1. THEOREM. (i) $D_\infty \models M = N \Leftrightarrow \mathcal{H}^* \vdash M = N \Leftrightarrow BT(M) \sim_\eta BT(N)$.
 (ii) $\mathcal{P}\omega \models M = N \Leftrightarrow BT(M) = BT(N)$.

The first result is due to Hyland and Wadsworth, the second to Hyland. We will only present the proof of (i) (see 7.16). See HYLAND [1976] or BARENDREGT [to appear] for the other proof.

7.2. COROLLARY (see 6.6 and 6.7 for a definition of the terms involved). *The equalities $Ax = Ay$, $Y_z = Y$, $J = I$ hold in D_∞ but cannot be proved algebraically, i.e. by conversion.*

PROOF. The equality in D_∞ follows by 6.6, 6.7 and the theorem. By the Church–Rosser theorem the equalities are not provable in λ . \square

7.3. DEFINITION. (In the sequel $\llbracket \cdot \rrbracket$ is $\llbracket \cdot \rrbracket^{D_\infty}$.) The $\lambda\Omega$ -calculus was introduced by Wadsworth as a tool for examining D_∞ . $\lambda\Omega$ -terms are defined by adding to the formation rules of terms (see 1.7): Ω is a term. The interpretation in D_∞ is extended to $\lambda\Omega$ -terms by setting $\llbracket \Omega \rrbracket = \perp$. Reduction for $\lambda\Omega$ -terms is ordinary reduction extended with the axioms $\lambda x\Omega \rightarrow \Omega$ and $\Omega P \rightarrow \Omega$. A $\lambda\Omega$ -term P has an Ω nf Q iff $P \rightarrow Q$ is an Ω reduction and Q has no subterm $(\lambda x.R)S$, $\lambda x\Omega$ or ΩR . Note that Ω reduction preserves the value in D_∞ since $\perp d = \perp$, hence also $\lambda x.\perp = \perp$. If P has a nf in the original sense, then P has an Ω nf, since replacements $\lambda x\Omega \rightarrow \Omega$ or $\Omega R \rightarrow \Omega$ decrease the length of a term. Böhm trees of $\lambda\Omega$ -terms are defined by letting $BT(\Omega) = \Omega$.

7.4. DEFINITION. Approximation.

(i) Let P, Q be $\lambda\Omega$ -terms. P approximates Q , notation $P \sqsubseteq Q$, iff $BT(P)$ results from $BT(Q)$ by replacing some subtrees by the tree Ω (e.g. $\lambda x.x\Omega \sqsubseteq \lambda x.IxK$).

(ii) Let M be a λ -term. P is an *approximate normal form* (anf) of M iff $P \sqsubseteq M$ and P is an Ω nf.

(iii) $A(M) = \{P \mid P \text{ is an anf of } M\}$.

(iv) M^k is the anf of M such that $BT(M^k)$ results from $BT(M)$ by replacing all labels at nodes of depth k by Ω and cancelling the deeper nodes. Note that $M^k \in A(M)$.

7.5. EXAMPLE. $\lambda f.f(\Omega) \sqsubseteq \lambda f.f((\lambda x.f(xx))(\lambda x.f(xx)))$ hence $\lambda f.f(\Omega)$ is an anf of the fixed point combinator Y (defined after 2.1). In fact

$$Y^k = \lambda f.f^k(\Omega), \quad A(Y) = \{\Omega(\leftarrow \lambda f.\Omega), \lambda f.f\Omega, \lambda f.f(f\Omega), \dots\}.$$

7.6. LEMMA. $P \sqsubseteq Q \Rightarrow \llbracket P \rrbracket \sqsubseteq \llbracket Q \rrbracket$.

PROOF. P is Q with some subterms replaced by Ω . Now $\llbracket \Omega \rrbracket = \perp$, the least element of D_∞ . The result follows since application and abstraction in D_∞ are monotonic (by their continuity). \square

The following theorem is quite useful for determining the value of λ -terms in D_∞ . The proof will be given in 7.18–7.24.

7.7. APPROXIMATION THEOREM (conjectured by Scott, proved by Hyland, and improved by Wadsworth). *For λ -terms M :*

$$\llbracket M \rrbracket = \bigsqcup \{ \llbracket P \rrbracket \mid P \in A(M) \}.$$

The same theorem holds in $\mathcal{P}\omega$ with the \bigsqcup replaced by \bigcup .

7.8. COROLLARY. $\llbracket M \rrbracket = \bigsqcup_k \llbracket M^k \rrbracket$.

PROOF. Let $P \in A(M)$, then $P \sqsubseteq M$, P Ω -nf. Now let all nodes in $\text{BT}(P)$ have depth $< k$. Then $\llbracket P \rrbracket \sqsubseteq \llbracket M^k \rrbracket$ and the result follows. \square

7.9. THEOREM. *The following are equivalent for $M \in \Lambda$:*

- (i) M is solvable,
- (ii) $\llbracket M \rrbracket \neq \perp$,
- (iii) M has a hnf.

PROOF. It may be assumed that M is closed (if not, consider $\lambda \vec{x}. M$ and note $\lambda d. \perp = \perp$).

(i) \Rightarrow (ii) Let M be solvable. Then for some \vec{N} , $\lambda \vdash M\vec{N} = I$. If $\llbracket M \rrbracket = \perp$, then $\llbracket I \rrbracket = \llbracket M\vec{N} \rrbracket = \perp \llbracket \vec{N} \rrbracket = \perp$, contradicting $\perp d = \perp$. So $\llbracket M \rrbracket \neq \perp$.

(ii) \Rightarrow (iii) Suppose M has no hnf. Then $A(M) = \{\Omega\}$, hence by 7.7, $\llbracket M \rrbracket = \perp$.

(iii) \Rightarrow (i) Suppose $M \rightarrow \lambda \vec{x}. x_i M_1 \cdots M_m$. Then by giving M enough arguments $N \equiv \lambda a_1 \cdots a_m. I$, M can be solved. \square

7.10. COROLLARY. D_∞ is sensible; hence $\lambda\eta + \mathcal{H}$ is consistent.

PROOF. By 7.9, M is unsolvable $\Rightarrow \llbracket M \rrbracket = \perp$; so $D_\infty \models \mathcal{H}$. Hence by 4.23, $D_\infty \models \lambda\eta + \mathcal{H}$. \square

Another consequence of the approximation theorem is a connection between the fixed point combinator and the least fixed point operator for complete lattices.

7.11. THEOREM (Tarski). *Let D be a complete lattice. Each continuous*

$f: d \rightarrow D$ has a fixed point. Moreover there is a $Y^* \in [[D \rightarrow D] \rightarrow D]$ such that Y^*f is the minimal fixed point of f .

PROOF. Let $Y^*f = \bigsqcup \{f^n(\perp) \mid n \in \omega\}$. Then Y^* is continuous by 4.8 and 4.5. Since $\perp \sqsubseteq f(\perp)$, so $f^n(\perp) \sqsubseteq f^{n+1}(\perp)$ $\{f^n(\perp) \mid n \in \omega\}$ is directed, hence Y^*f is a fixed point of f . If $f(x) = x$, then since $\perp \sqsubseteq x$, $f(\perp) \sqsubseteq f(x) \sqsubseteq x$, etc., so $Y^*f \sqsubseteq x$. \square

Let Y_{Tarski} be the element of D_∞ corresponding to the fixed point operator Y^* and let $Y_{\text{Curry}} = \llbracket Y \rrbracket$, with Y defined after 2.1.

7.12. THEOREM (Park). *In Scott's model D_∞ , $Y_{\text{Tarski}} = Y_{\text{Curry}}$.*

PROOF. By 7.8 and 7.5,

$$Y_{\text{Curry}}d = (\bigsqcup_k \llbracket \lambda f. f^k(\Omega) \rrbracket)d = \bigsqcup_k d^k(\perp) = Y_{\text{Tarski}}d$$

by definition. The result now follows by extensionality. \square

Results analogous to 7.11 and 7.12 hold for $\mathcal{P}\omega$.

In order to prove the characterization of equality in D_∞ , the following definition is needed.

- 7.13. DEFINITION.** (i) $M \sqsubseteq_k N$ iff $M^k \sqsubseteq N^k$;
 (ii) $M <_k N$ iff $\exists M', N' [\lambda \eta \vdash M = M', \lambda \eta \vdash N = N' \text{ and } M' \sqsubseteq_k N']$;
 (iii) $M < N$ iff $\forall k M <_k N$.

Note that if $\text{BT}(M) \sim_\eta \text{BT}(N)$, then $M < N$ and $N < M$. The converse follows from 7.14 and 7.16. Also note $M^k < M$. $<$ is transitive, since if $M_1^k \sqsubseteq N_1^k$ and $N_2^k \sqsubseteq L_2^k$, where $\lambda \eta \vdash N_1 = N_2$, then by some $\beta\eta$ -conversion M_1 becomes M_2 such that $M_2^k \sqsubseteq N_2^k \sqsubseteq L_2^k$.

7.14. THEOREM. $M < N \Rightarrow \llbracket M \rrbracket \sqsubseteq \llbracket N \rrbracket$.

The proof is given in 7.25–7.27.

7.15. COROLLARY. $\text{BT}(M) \sim_\eta \text{BT}(N) \Rightarrow \llbracket M \rrbracket = \llbracket N \rrbracket$.

PROOF. By the remark following 7.13. \square

7.16. THEOREM (Hyland; Wadsworth). *The following are equivalent for $M, N \in \Lambda$.*

- (i) $\text{BT}(M) \sim_\eta \text{BT}(N)$,
- (ii) $D_\infty \models M = N$,
- (iii) $\mathcal{H}^* \vdash M = N$.

PROOF. (i) \Rightarrow (ii) is 7.15.

(ii) \Rightarrow (iii) by 5.9 since $D_\infty \models \mathcal{H}$ (7.10).

(iii) \Rightarrow (i) is 6.8. \square

Note that $\text{Th}(D_\infty) \subset \mathcal{H}^* \Rightarrow \text{Th}(D_\infty) = \mathcal{H}^*$ is not obvious (as would have been the case in ordinary model theory): since the language of the λ -calculus does not contain logical operators, $\text{Th}(\mathcal{M})$ is not necessarily a complete set of sentences.

7.17. COROLLARY. (i) $\text{Th}(D_\infty) = \mathcal{H}^*$; hence for every D , D_∞ satisfies the same set of equations,

(ii) D_∞^0 is algebraically simple.

PROOF. (i) Immediate.

(ii) $D_\infty^0 = \mathcal{M}^0(\text{Th}(D_\infty^0)) = \mathcal{M}^0(\mathcal{H}^*)$ hence the result follows from 5.11. \square

In order to prove the approximation theorem the following indexed $\lambda\Omega$ -calculus was introduced by Hyland and Wadsworth.

7.18. DEFINITION. The set of *indexed* $(\lambda\Omega)$ -terms is defined by adding to the formation rules of the $\lambda\Omega$ -terms: if M is an indexed $\lambda\Omega$ -term, so is $(M)^p$ for all $p \in \omega$. The interpretation of $\lambda\Omega$ -terms in D_∞ is extended to indexed terms by adding the clause $\llbracket (M)^p \rrbracket_p = (\llbracket M \rrbracket_p)_p$. Thus the superscripts p are considered as the projections $\Phi_{\infty, p} : D_\infty \rightarrow D_p$.

If M is an indexed term, M^* is obtained from M by leaving out all superscripts. Note that $\llbracket M \rrbracket \subseteq \llbracket M^* \rrbracket$. A *completely* indexed term M is such that each subterm occurrence N of M^* has an index in M (i.e. occurs as part of $(N)^p$ in M).

7.19. DEFINITION. The reduction relation on $\lambda\Omega$ -terms is extended to indexed $\lambda\Omega$ -terms by adding the axioms

$$\begin{aligned} \lambda x. \Omega^p &\rightarrow \Omega^0; & \Omega^p M &\rightarrow \Omega^0; & (\lambda x. M)^{p+1} N &\rightarrow (M[x/N^p])^p; \\ (\lambda x. M)^0 N &\rightarrow (M[x/\Omega^0])^0; & (M^p)^q &\rightarrow M^{\min(p, q)} \end{aligned}$$

and the rule $M \rightarrow N \Rightarrow M^p \rightarrow N^p$. An indexed term M has a nf if for some N , $M \rightarrow N$ and N^* is in Ω nf.

7.20. LEMMA. *Let M, N be indexed terms, and $M \rightarrow N$ as indexed terms. Then*

- (i) $N^* \sqsubseteq M^*$.
- (ii) $\llbracket M \rrbracket = \llbracket N \rrbracket$.

PROOF. (i) The approximation comes in at reductions like $(\lambda x. x)^0 N \rightarrow \Omega^0$.
(ii) By 4.19 and 4.15(i), (v). \square

7.21. DEFINITION. (i) An *indexing* I on M is a mapping that assigns to each subterm occurrence of M an index. M' is the resulting completely indexed term.

- (ii) $\tau(M) = \{M' \mid I \text{ indexing on } M\}$.

7.22. LEMMA. *Let M be a λ -term. Then $\llbracket M \rrbracket = \bigsqcup \{\llbracket N \rrbracket \mid N \in \tau(M)\}$.*

PROOF. By induction on the structure of M , using 4.15(iii). \square

7.23. LEMMA. *Each completely indexed term has a nf.*

PROOF. M has a p -redex iff $(\lambda x. P)^p Q$ occurs in M . The *order* of M is the maximal p such that M has a p -redex. By induction on the order p of M , it is shown that M has a nf.

$p = 0$: contractions like $(\lambda x. P)^0 Q \rightarrow (P[x/\Omega^0])^0$, $\lambda x. \Omega^p \rightarrow \Omega^0$, $\Omega^p M \rightarrow \Omega^0$ and $(M^p)^q \rightarrow M^{\min(p, q)}$ decrease the length of a term, hence each term of order 0 has a nf.

$p = n + 1$: replacing the rightmost $n + 1$ redex $(\lambda x. P)^{n+1} Q$ by $(P[x/Q^n])^n$ and then replacing terms $(Q^n)^q$ by $Q^{\min(n, q)}$ results in a term with one less occurrence of a p -redex. (Prime example (some indices are omitted):

$$\begin{aligned} (\lambda ab. baa)^{n+1}((\lambda x. x^{n+1} R)^{n+1}(\lambda z. z)^{n+2}) &\rightarrow (\lambda ab. baa)^{n+1}, \\ &\rightarrow (\lambda ab. baa)^{n+1}(((\lambda z. z)^n)^{n+1} R) \rightarrow (\lambda ab. baa)^{n+1}((\lambda z. z)^n R). \end{aligned}$$

After a finite number of steps the term is reduced to one with order n and the induction hypothesis applies. \square

7.24. PROOF OF 7.7. $\llbracket M \rrbracket = \bigsqcup \{\llbracket N \rrbracket \mid N \in \tau(M)\} = \bigsqcup \{\llbracket L \rrbracket \mid \exists N \in \tau(M) L \text{ nf of } N\} \sqsubseteq \bigsqcup \{\llbracket L^* \rrbracket \mid \exists N \in \tau(M) L \text{ nf of } N\} \sqsubseteq \bigsqcup \{\llbracket N \rrbracket \mid N \in A(M)\} \sqsubseteq \llbracket M \rrbracket$.

The five (in)equalities follow respectively from 7.22, 7.20(ii) and 7.23, $\llbracket L \rrbracket \sqsubseteq \llbracket L^* \rrbracket$, $L^* \in A(M)$ since by 7.20(i) $L^* \sqsubseteq N^* = M$, and from $N \in A(M) \Rightarrow \llbracket N \rrbracket \sqsubseteq \llbracket M \rrbracket$ (by 7.6). \square

Now Hyland's proof of 7.14 will be presented.

7.25. LEMMA. *Let $P \neq \Omega$ be an Ω nf and $P < Q$. Then $BT(P)$, $BT(Q)$ are top mergeable, $\lambda\eta \vdash P = \lambda x_1 \cdots x_n. x_i P_1 \cdots P_m$ and $\lambda\eta \vdash Q = \lambda x_1 \cdots x_n. x_i Q_1 \cdots Q_m$ say, and $P_1 < Q_1, \dots, P_m < Q_m$; the P_1, \dots, P_m are in Ω nf.*

PROOF. Only η -reductions affect the Böhm trees. Therefore, since $P < Q$, after some η -changes the tops of $BT(P)$, $BT(Q)$ are the same, i.e. they are top mergeable. The P_i 's are either already part of P or a variable created by an η -expansion; therefore they are in Ω nf. Since $P <_{k+1} Q$, it follows that $P_i <_k Q_i$ for all k ; hence $P_i < Q_i$. \square

7.26. LEMMA. *Let P be an Ω nf. Then $P < N \Rightarrow \llbracket P \rrbracket \subseteq \llbracket N \rrbracket$.*

PROOF. By induction on the structure of P ,

Case 1. $P \equiv \Omega$. Then we are done.

Case 2. $P \equiv x$. Then, using $x <_1 N$, $\lambda \vdash N = \lambda y_1 \cdots y_n. x N_1 \cdots N_n$. By 7.22 it is sufficient to show that for any indexing I of x $\llbracket x \rrbracket^I \subseteq \llbracket N \rrbracket$. This is done by induction on $k = I(x)$.

If $k = 0$, then $\llbracket (x)^0 \rrbracket = \llbracket \lambda y_1 \cdots y_n. (x)^0 \Omega \cdots \Omega \rrbracket \subseteq \llbracket N \rrbracket$ since in D_x $x_0 = x_0 \cdot \perp = \lambda y. x_0$ by 4.19(ii).

If $I(x) = k + 1$, then $\llbracket (x)^{k+1} \rrbracket = \llbracket \lambda y_1 \cdots y_n. (x)^{k+1} (y_1)^k \cdots (y_n)^{k+1-n} \rrbracket$ by 4.19(i) (and 4.14(i)). Since $x < N$, for all i , $1 \leq i \leq n$, $y_i < N_i$, hence by the induction hypothesis $\llbracket y_i^{k+1-i} \rrbracket \subseteq \llbracket N_i \rrbracket$. So we have $\llbracket (x)^{k+1} \rrbracket \subseteq \llbracket N \rrbracket$.

Case 3. $P \equiv \lambda x_1 \cdots x_m. x P_1 \cdots P_n$, P_1, \dots, P_n Ω nf's. Then since $P < N$ for some P', N' with $\lambda\eta \vdash P' = P$, $\lambda\eta \vdash N' = N$ one has $P' \equiv \lambda x_1 \cdots x_p. x P'_1 \cdots P'_q$, $N' \equiv \lambda x_1 \cdots x_p. x N'_1 \cdots N'_q$ and $P'_1 < N'_1, \dots, P'_q < N'_q$.

By the induction hypothesis $\llbracket P_i \rrbracket \subseteq \llbracket N \rrbracket, \dots$ hence $\llbracket P \rrbracket = \llbracket P' \rrbracket \subseteq \llbracket N' \rrbracket = \llbracket N \rrbracket$. \square

7.27. PROOF OF THEOREM 7.14. Suppose $M < N$. Then $\forall k M^k < N$, hence by 7.26, $\forall k \llbracket M^k \rrbracket \subseteq \llbracket N \rrbracket$. Therefore $\llbracket M \rrbracket = \bigsqcup \llbracket M^k \rrbracket \subseteq \llbracket N \rrbracket$. \square

References

BARENDREGT, H.P.

- [1975] Normed uniformly reflexive structures, in: *λ -Calculus and Computer Science*, edited by C. Böhm, Lecture Notes in Computer Science, Vol. 37 (Springer, Berlin) pp. 272–286.

- [1977] *The Lambda Calculus, its Syntax and Semantics* (North-Holland, Amsterdam) to appear.
- BÖHM, C.
 [1968] Alcune proprietà della forma β - η -normali del λ -K-calcolo, Pubbl. IAC-CNR n. 696, Roma.
- BRUIJN, N.G. DE
 [1972] Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem, *Indag. Math.*, **34** (5), 381-392.
- CHURCH, C.
 [1941] *The Calculi of Lambda-Conversion* (Princeton University Press, Princeton, RI).
- CURRY, H.B. and R. FEYS
 [1958] *Combinatory Logic*, Vol. I (North-Holland, Amsterdam).
- CURRY, H.B., R. HINDLEY and J. SELDIN
 [1972] *Combinatory Logic*, Vol. II (North-Holland, Amsterdam).
- HINDLEY, R., B. LERCHER and J. SELDIN
 [1972] *Introduction to Combinatory Logic* (Cambridge University Press, Cambridge).
- HYLAND, J.M.E.
 [1976] A syntactic characterization of the equality in some models of the λ -calculus, *J. London Math. Soc.*, **12** (2), 361-370.
- MANN, C.R.
 [1975] The connection between equivalence of proofs and cartesian closed categories, *Proc. London Math. Soc.*, **31** (3), 289-310.
- NAKAJIMA, R.
 [1975] Infinite normal forms for λ -calculus, in: *λ -Calculus and Computer Science*, edited by C. Böhm, Lecture Notes in Computer Science, Vol. 37 (Springer, Berlin) pp. 62-82.
- PLOTKIN, G.
 [1972] A set-theoretical definition of application, School of A.I., Memo MIP-R-95, Edinburgh.
 [1974] The λ -calculus is ω -incomplete, *J. Symbolic Logic*, **39** (2), 313-317.
- SCOTT, D.
 [1972] Continuous lattices, in: *Toposes, Algebraic Geometry and Logic*, edited by F.W. Lawvere, Lecture Notes in Mathematics, Vol. 274 (Springer, Berlin) pp. 97-136.
 [1975a] Lambda calculus and recursion theory, in: *Proceedings of the Third Scandinavian Logic Symposium*, edited by S. Kanger (North-Holland, Amsterdam) pp. 154-193.
 [1975b] Combinators and classes, in: *λ -Calculus and Computer Science*, edited by C. Böhm, Lecture Notes in Computer Science, Vol. 37 (Springer, Berlin) pp. 1-26.
 [1975c] Some philosophical issues concerning theories of combinators, in: *λ -Calculus and Computer Science*, edited by C. Böhm, Lecture Notes in Computer Science, Vol. 37 (Springer, Berlin) pp. 346-366.
 [1976] Data types as lattices, *SIAM J. Comput.*, **5** (3), 522-587.
- TROELSTRA, A.S., editor
 [1973] *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, Lecture Notes in Mathematics, Vol. 344 (Springer, Berlin).
- WADSWORTH, C.P.
 [1976] The relation between lambda-expressions and their denotations, *SIAM J. Comput.*, **5** (3), 488-521.